



UNIVERSIDADE DE CABO VERDE

RELATÓRIO DO ESTÁGIO

LICENCIATURA EM TECNOLOGIAS DE INFORMAÇÃO E
COMUNICAÇÃO

SISTEMA DE GESTÃO UNIVERSITÁRIA
IMPLEMENTAÇÃO E CUSTOMIZAÇÃO DO MÓDULO
FINANCEIRO

O caso da Universidade de Cabo Verde

FLÁVIO DE JESUS VIEIRA SEMEDO

N^o: 2813

CAMPUS PALMAREJO, DEZEMBRO, 2009

FLÁVIO DE JESUS VIEIRA SEMEDO

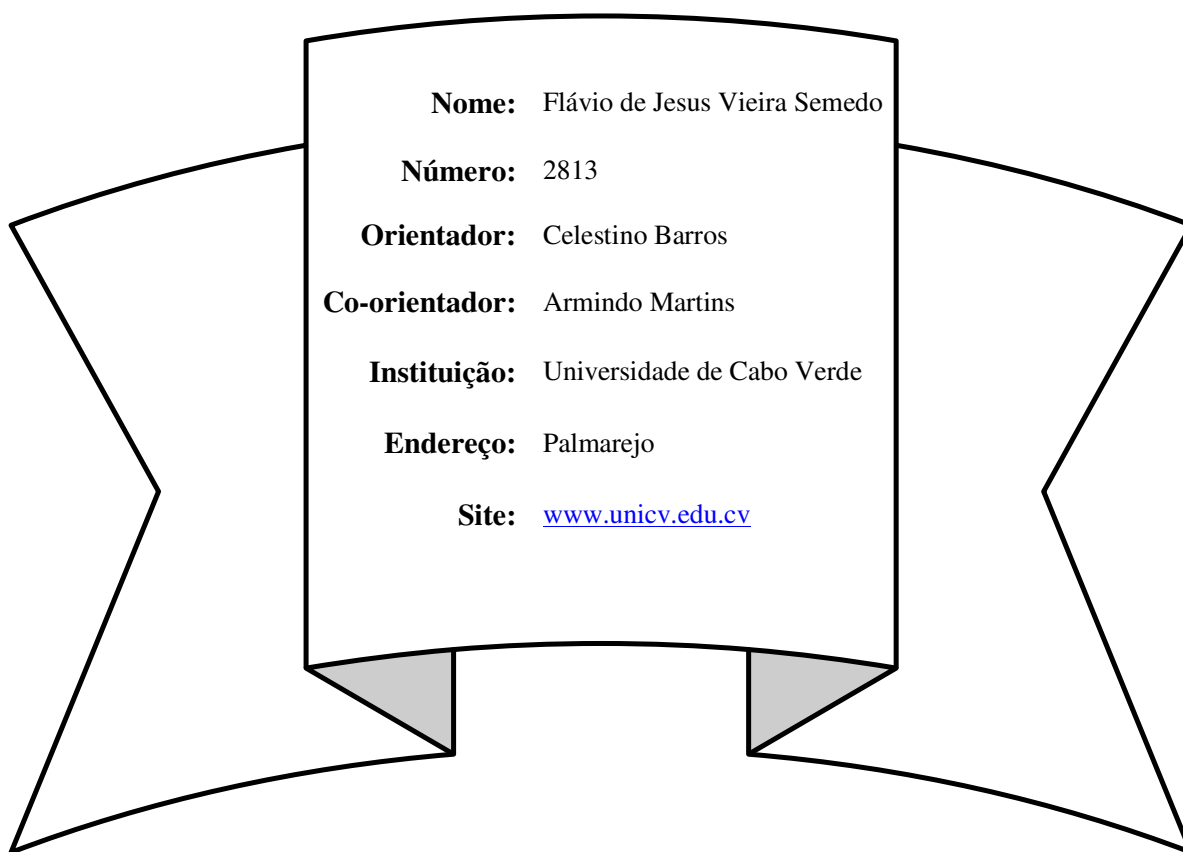
N.º: 2813

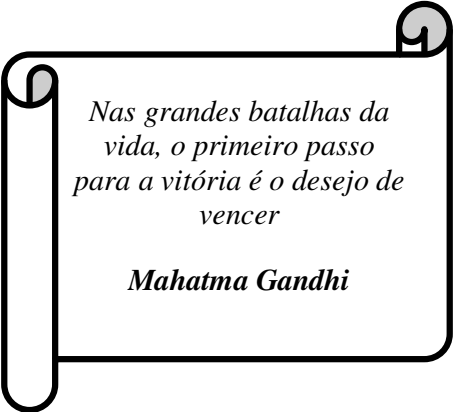
SISTEMA DE GESTÃO UNIVERSITÁRIA
IMPLEMENTAÇÃO E CUSTOMIZAÇÃO DO
MÓDULO FINANCEIRO

Relatório do Projecto do Estágio Realizado na Universidade de Cabo Verde

Relatório do projecto/estágio apresentado na Universidade de Cabo Verde (UNICV) para obtenção do grau de Licenciatura em Tecnologias de Informação e Comunicação, sob a orientação do Dr. Celestino Lopes Barros e co-orientação do Eng.º Armino Martins

DEZEMBRO, 2009





*Nas grandes batalhas da
vida, o primeiro passo
para a vitória é o desejo de
vencer*

Mahatma Gandhi

Resumo

O presente trabalho consiste no relatório de estágio como parte conclusivo da licenciatura em Tecnologias de Informação e Comunicação. A temática em estudo centraliza-se na gestão da Universidade de Cabo verde (Uni-CV), onde, desde a sua instalação, se procurou estar cada vez mais preparado no sentido de conseguir um crescimento contínuo no que concerne à utilização das Novas Tecnologias de Informação e Comunicação, que constituem suporte inevitável para gerir as informações que se têm aumentado no seio desta instituição.

Este projecto visa a Implementação de um Sistema de Gestão Universitária baseado em *softwares* livres. Especificamente, o objecto de estudo deste projecto consiste na Implementação de um Módulo Financeiro na Universidade, com o intuito de melhorar os serviços de cobrança de propinas.

Palavras-chave: Sistema de Informação, *Softwares* Livres, Sistema de Gestão Universitária, Implementação do Módulo Financeiro.

Agradecimentos

A minha gratidão vai antes de mais para o meu orientador, Dr. Celestino Barros, que fez com que este trabalho pudesse crescer saudável. A ele devo todo o apoio e confiança ilimitados que me deixaram sempre entusiasmado e livre para todas as escolhas, sabendo, de antemão, que pudesse contar sempre com ele e com o seu saber. Esta gratidão é extensiva também ao meu co-orientador, Engenheiro Armindo Martins, pela atenção e disponibilidade que demonstrou, sempre que solicitado.

Alguns familiares, colegas e amigos foram, em vários momentos, de um ou outro modo, apoios indispensáveis. A todos o meu agradecimento:

Aos meus pais e meus irmãos, pelo apoio incondicional durante todos estes anos, por terem depositado em mim o espírito de luta. Para eles, o meu muito obrigado, pois a eles devo tudo o que sou hoje.

Aos meus colegas do curso, agradeço as valiosas contribuições que sugeriram no estágio e também por terem discutido comigo, para o meu proveito, partes deste projecto.

Por último, não menos importante, a todos meus amigos pela amizade e pelo espírito de humor que nos uniu durante estes anos. E também a todos aqueles que de uma forma directa ou indirecta me apoiaram com o seu bom astral e pensamentos positivos.

Índice de Tabelas

TABELA 1: <i>SOFTWARES</i> ANALISADOS-----	29
TABELA 2: DESCRIÇÃO DOS ACTORES-----	35
TABELA 3: DESCRIÇÃO DE CASO DE UTILIZAÇÃO-----	35

Índice de Figuras

FIGURA 1: CAMADAS	20
FIGURA 3: QUOTAS DE MERCADO DE SERVIDORES WEB	22
FIGURA 4: ARQUITECTURA DE FUNCIONAMENTO DE SGU	28
FIGURA 5: CASO DE USO PARA PROCESSAMENTO DE TÍTULOS A RECEBER.....	34
FIGURA 6: MODELO ENTIDADE RELACIONAMENTO (MER)	37
FIGURA 7: DIAGRAMA DE ACTIVIDADE PARA CRIAR TÍTULOS A RECEBER	38
FIGURA 8: DIGRAMA DE SEQUÊNCIA PARA CRIAR TÍTULOS A RECEBER	39
FIGURA 9: DIAGRAMA DE ACTIVIDADE PARA COBRANÇA DE PROPINAS.....	40
FIGURA 10: DIAGRAMA DE SEQUÊNCIA PARA COBRANÇA DE PROPINAS	41
FIGURA 11: PÁGINA DE ACESSO DO SGU	43
FIGURA 12: MÓDULO FINANCEIRO.....	43
FIGURA 13: CRIAR TÍTULOS	44
FIGURA 14: INFORMAÇÕES DO LANÇAMENTO.....	44
FIGURA 15: FECHAR TÍTULO	45

Lista de Abreviatura

BSD	<i>Berkeley Software Distribution</i>
ERP	<i>Enterprise Resource Planning</i>
GNU	<i>General Public License</i>
GPL	<i>General Public License</i>
GTK	<i>GNU Image Manipulation Program ToolKit</i>
HTTP	<i>Hyper Text Transfer Protocol</i>
HTML	<i>Hyper Text Markup Language</i>
LDAP	<i>Lightweight Directory Access Protocol</i>
PHP	<i>PHP HiperText Preprocessor</i>
PDF	<i>Portable Document File</i>
SAGU	Sistema Aberto de Gestão Unificado
SGU	Sistema de Gestão Universitária
SI	Sistemas de Informações
SOLIS	Cooperativa de Soluções Livres
SSL	<i>Secure Socket Layer</i>
TI	Tecnologias de Informação
TLS	<i>Transport Layer Security</i>
UML	<i>Unified Modelling Language</i>
Uni-CV	Universidade de Cabo Verde

ÍNDICE

1.	INTRODUÇÃO	1
1.1.	ENQUADRAMENTO	1
1.2.	CONTEXTO	1
1.3.	APRESENTAÇÃO DA UNIVERSIDADE DE CABO VERDE	2
1.3.1.	<i>Missão da Uni-CV</i>	3
1.4.	SITUAÇÃO PROBLEMÁTICA	3
1.5.	OBJECTIVOS DO PROJECTO	4
1.6.	JUSTIFICATIVA DO PROJECTO	4
1.7.	METODOLOGIA UTILIZADA	5
1.8.	ESTRUTURA DO RELATÓRIO	6
2.	SISTEMA DE INFORMAÇÃO NAS UNIVERSIDADES	7
2.1.	SISTEMAS DE INFORMAÇÕES	7
2.2.	SISTEMA DE GESTÃO INTEGRADO	8
2.3.	FASES DA CONCEPÇÃO DO SISTEMA DE INFORMAÇÃO	8
2.4.	SISTEMA ABERTO DE GESTÃO UNIFICADO	10
2.4.1.	<i>Histórico SAGU</i>	10
2.4.2.	<i>Características do SAGU</i>	11
2.4.3.	<i>Módulos do SAGU</i>	12
3.	TECNOLOGIAS E FERRAMENTAS UTILIZADAS	16
3.1.	TECNOLOGIAS OPEN SOURCE	16
3.1.1.	<i>Vantagens e Desvantagens dos Softwares Livres</i>	17
3.2.	FERRAMENTAS UTILIZADAS	18
3.2.1.	<i>Sistema Operativo Ubuntu</i>	19
3.2.2.	<i>Framework de Desenvolvimento (Miolo)</i>	19
3.2.3.	<i>Servidor Apache</i>	21
3.2.4.	<i>Base de Dados PostgreSQL</i>	22
3.2.5.	<i>Gestão de Base de Dados PostgreSQL: Pgadmin</i>	23
3.2.6.	<i>Gerador de Relatórios (Agata Report)</i>	23
3.2.7.	<i>PoEdit</i>	24
3.3.	LINGUAGENS UTILIZADAS	24
3.3.1.	<i>UML (Unified Modelling Language)</i>	24
3.3.2.	<i>HTML (HyperText Markup Language)</i>	24
3.3.3.	<i>PHP (HyperText Preprocessor)</i>	25
4.	IMPLEMENTAÇÃO DO MÓDULO FINANCEIRO	26
4.1.	PROPOSTA DO SISTEMA DE GESTÃO UNIVERSITÁRIA	26
4.1.1.	<i>Arquitectura do SGU</i>	27
4.2.	RAZÕES DA ESCOLHA DO SAGU	28
4.3.	IMPLEMENTAÇÃO DO MÓDULO FINANCEIRO	30
4.4.	ACTIVIDADES PLANEADAS	30
4.5.	DIFICULDADES SENTIDAS DURANTE O ESTÁGIO	31
4.6.	DESCRIÇÃO DOS REQUISITOS	31
4.6.1.	<i>Levantamento dos Dados</i>	31
4.7.	DESCRIÇÃO DA SITUAÇÃO ACTUAL	31
4.7.1.	<i>Análise de Requisitos</i>	31
4.7.2.	<i>Solução Proposta</i>	32
4.8.	REQUISITOS DE HARDWARE E SOFTWARE	32
4.8.1.	<i>Requisitos de Instalação</i>	33
4.9.	PROJECTO CONCEPTUAL	33
4.9.1.	<i>Descrição dos Casos de Utilização</i>	34
4.10.	PROJECTO LÓGICO	36
4.11.	DIAGRAMA DE ACTIVIDADE	38
4.11.1.	<i>Processo Criação de Títulos a Receber</i>	38

4.11.2.	<i>Processo para a Pagamento de Propinas</i>	39
4.12.	TESTES	41
4.13.	APRESENTAÇÃO DA IMPLEMENTAÇÃO	42
5.	CONCLUSÃO E RECOMENDAÇÕES	46
	REFERÊNCIAS BIBLIOGRÁFICAS	48
	ANEXOS	50

1. Introdução

1.1. Enquadramento

Este projecto é elaborado no âmbito da realização do estágio na área dos Serviços Técnicos da Universidade de Cabo Verde, no curso de Tecnologias de Informação e Comunicação, com o intuito de obtenção do grau de licenciatura. O tema é “Sistema de Gestão Universitária: Implementação e Customização do Módulo Financeiro.” Nele, estão tratados os aspectos relevantes para a implementação de um Módulo Financeiro na Universidade, que visa proporcionar a melhoria dos serviços de cobrança das propinas.

1.2. Contexto

Segundo (Attila, 2004), um dos recursos mais valiosos numa organização em qualquer ramo de negócio é o uso da informação e tecnologia de informação. Deste modo, as organizações para serem inteligentes, precisam de disponibilizar produtos de qualidades, praticar bom atendimento, adequar boa política de negócio e estar atentas às mutações do mercado (Alsene, 1999).

A Universidade de Cabo Verde tem implementado profundas mudanças desde a sua instalação. Nesta perspectiva, as suas unidades constituintes devem também acompanhar esse processo de mudança e transformação, para uma maior harmonia e alinhamento de todos os seus colaboradores em relação aos objectivos e cumprimento da sua missão.

Face a este cenário, a Universidade de Cabo Verde sentiu a necessidade de melhorar os serviços prestados e racionalizar os seus recursos, fazendo assim a optimização da sua gestão.

Ao integrar os serviços através de um sistema de informações, a Universidade estará apostando fortemente numa gestão eficaz e transparente, em busca de maior produtividade e melhoria dos serviços prestados, para atender cada vez melhor as expectativas dos seus alunos e colaboradores. Por esta razão, a implementação do Sistema de Gestão Universitária (SGU) revelou-se um caminho adequado para alcançar os objectivos traçados pela mesma.

O SGU visa a melhoria no desempenho da gestão universitária, integrando as áreas académicas, administrativa e financeiras. A aplicação é fundamentada na abordagem *Open Source*, possui uma estrutura modular e uma base de dados integrado, com o acesso dentro de um ambiente Web (intranet, internet).

A implementação do SGU requer ainda um conhecimento pormenorizado da Universidade, das suas actividades e dos seus problemas administrativos. Um sistema de informações que possibilite a interacção entre as áreas administrativas, financeira e a área académica propicia melhor controlo sobre dissipação e melhoria no processo de decisões em relação aos investimentos no processo ensino-aprendizagem.

1.3. Apresentação da Universidade de Cabo Verde

Entre os diversos fenómenos sociais registados na sociedade Cabo-Verdiana após a independência, a democratização da educação afigura-se um dos que se manifestou com maior evidência no processo de desenvolvimento de Cabo Verde.

O primeiro passo para a implantação do Ensino Superior em Cabo Verde deu-se em 1979, com a criação do Curso de Formação de Professores do Ensino Secundário com o objectivo de dotar o País de professores capacitados para o prosseguimento dos objectivos em matéria de educação.

Desde então o Ensino Superior em Cabo Verde vem sofrendo diversas transformações em busca de excelência na matéria de educação com a cooperação com várias Instituições de Ensino Superior, o que originou a criação de novos cursos e centros de investigações em várias vertentes de ensino.

A intensificação da procura do ensino superior, seguido de novas exigências de uma sociedade mais reivindicadora, tem levado o país a dinâmicas adaptativas do seu modelo de ensino superior, caracterizado pela reorganização das instituições públicas até então existentes.

A necessidade de acompanhar a dinâmica interna e externa (a evolução científica e tecnológica e os novos paradigmas do mercado e do mundo globalizante) veio exigir de todos

os cabo-verdianos a mudança de atitude frente aos novos desafios que se avizinham. Nesse contexto, Em 2004 foi criada através do Decreto-lei nº 31/2004, a Comissão Nacional para a Instalação da Universidade de Cabo Verde (CNI-UniCV), tendo por missão programar e desenvolver as actividades necessárias à instalação da Universidade Pública de Cabo Verde.

Em 21 de Novembro de 2006, instala-se, realmente a Universidade de Cabo Verde visando contribuir de uma forma decisiva para o desenvolvimento durável de CABO VERDE nos domínios científico, tecnológico, económico, social e cultural.¹

1.3.1. Missão da Uni-CV

A missão da Universidade consiste em contribuir para o desenvolvimento de Cabo Verde nos domínios científico, tecnológico, económico, social e cultural. Assim, a Universidade de Cabo Verde procurará ser um referencial de excelência académica em todas as áreas de pertinência para o desenvolvimento do País. Com efeito, ela deve estar comprometida com as grandes causas da sociedade.

1.4. Situação Problemática

De acordo com a entrevista realizada com um dos técnicos responsáveis pelo serviço financeiro da Uni-CV, verificou-se que o sistema de informação actual utilizado pela Universidade de Cabo Verde não responde com eficácia e eficiência às necessidades actuais da mesma. Segundo ele, a Uni-CV enfrenta grandes problemas no que tange essencialmente o controlo e cobrança de propinas. Este processo era realizado de forma tradicional, pelo que são utilizados cadernetas para cobrança de propinas, o que faz com que este processo torna-se burocrático, e pode gerar erros, perdendo assim a fiabilidade dos dados. Uma outra dificuldade a ser constatada é a descentralização de informações entre os Serviços Académicos e Financeiros, o que torna difícil obter relatórios financeiros para a obtenção de documentação (declarações, lista discriminada, entre outros) do aluno, por não existir um

¹ Estudos de casos hoje confirmam a importância do investimento na educação, nomeadamente no ensino superior, como um factor primordial para o desenvolvimento e determinante no crescimento individual, da produtividade, da participação política e do reforço da sociedade civil e da democratização da sociedade, para além de constituir um espaço de liberdade e de formação de competências para todos os níveis do sistema educativo. Disponível no site www.unicv.edu.cv

sistema informatizado de gestão integrado que permite apresentar em tempo real a situação actual deste na Universidade.

1.5. Objectivos do Projecto

O objectivo principal deste projecto consiste em implementar um Sistema de Gestão Universitária baseado em Tecnologias *Open Source*, com o foco principal na implementação do Módulo Financeiro e terá como resultado final responder às necessidades da Universidade de Cabo Verde.

Desta forma, os objectivos específicos são os que se enunciam de seguida:

- Fazer um levantamento das necessidades actuais da área financeira da Universidade de Cabo Verde;
- Estudar uma solução que seja capaz de não só gerir os serviços financeiros, como também propor a gestão das demais áreas administrativas da Universidade de Cabo Verde;
- Estudar as tecnologias e ferramentas que serão utilizados na implementação e customização do Módulo Financeiro para a cobrança de propinas;
- Preparar o protótipo de sistema a ser customizado no Módulo Financeiro para a cobrança de propinas;
- Definir o nível de acesso, segurança e controlo deste Módulo financeiro;
- Programar as customizações planeadas para a implementação do Módulo Financeiro em referência.
- Fazer o teste do Módulo Financeiro integrado com outros módulos do sistema para eliminação de erros.

1.6. Justificativa do Projecto

A implementação do SGU na Uni-CV justifica-se pela capacidade de apoio ao processo de gestão e à tomada de decisão. Actualmente a Uni-CV apresenta uma grande quantidade de

processos devido à demanda que está a crescer cada vez mais no seio da mesma. Para dar resposta a estas demandas a Universidade necessitará de um sistema de gestão para o tratamento dessas informações de forma eficaz e eficiente.

O SGU apresenta as seguintes vantagens para a gestão da Uni-CV:

- Uma base de dados centralizada, o que traz maior fiabilidade às informações e o acesso em tempo real;
- Melhor interacção entre as áreas administrativas da Universidade;
- Diminuição da carga de actividades, o que se traduz pela eliminação das redundância de actividades;
- Redução do tempo total de andamento dos processos;
- Rapidez na tomada de decisão;

1.7. Metodologia Utilizada

Antes de iniciar o projecto, procedeu-se à fase de recolha e análise de informações, na qual foi efectuado um estudo das necessidades actuais da Universidade, com o objectivo de se encontrarem ideias acerca das funcionalidades que gostaríamos de ver implementadas e quais as que melhor se adaptavam aos requisitos do projecto. Em seguida foi efectuado um estudo conceptual dos requisitos do sistema a ser adaptado, segundo as necessidades encontradas.

Para a concepção deste projecto foi preciso o envolvimento dos colaboradores da Universidade de Cabo Verde durante a fase de elaboração do projecto até a sua execução. A implementação do SGU não teria sucesso sem que as pessoas envolvidas tivessem consciência clara das consequências das suas acções, o que demonstrou que a gestão universitária é um factor crítico neste processo. Neste âmbito a Implementação do SGU foi realizada por uma equipa dividida por módulos, sendo que a integração destes módulos possibilitou o fluxo dos processos dentro do sistema.

Em relação à técnica da pesquisa, foi efectuado uma análise preliminar dos *softwares* que melhor se adequem às exigências da Universidade, examinando as suas características, as ferramentas de uso e a linguagem utilizada para a sua implementação. Foi apontado como o

melhor *software* livre o SAGU (Sistema Aberto de Gestão Unificado) dentre os *softwares* estudados (TesEscola, AdX, UniMestre, SophiA, Corpore RM Education al), pelo que foi empregue uma outra técnica referente ao estudo de caso, onde procurou-se analisar as medidas adoptadas pela Solis² na concepção do sistema SAGU, relativamente às tecnologias e ferramentas utilizadas. O estudo baseou-se também nas revisões da literatura que retrata o tema de Sistemas de Gestão Integrado, baseando-se em pesquisas realizadas através da internet, de livros, artigos, entre outros documentos. Foi utilizada a técnica de entrevista não estruturada, com perguntas abertas para a recolha de informações relativamente às áreas Financeira e Académica.

1.8. Estrutura do Relatório

O presente relatório encontra-se estrutura-se em cinco capítulos, constituindo o produto principal do estágio da Licenciatura em Tecnologias de Informação e Comunicação. O primeiro capítulo evidencia a estrutura do mesmo. Foi apresentado neste capítulo a apresentação da organização onde foi realizado o estágio, a descrição do objectivo do trabalho, a justificativa para o seu desenvolvimento, o método de trabalho a ser empregado. O capítulo 2 contempla a revisão bibliográfica a respeito do tema escolhido, focando os principais assuntos envolvidos, apresentando a perspectiva histórica do sistema de informação e sistemas integrados e em seguida faz referência ao sistema escolhido para ser customizado e implementado na organização em estudo apresentando as suas principais características. Nesta perspectiva o terceiro capítulo insere-se no âmbito das tecnologias e ferramentas utilizadas. Assim sendo, é apresentada de uma forma breve a descrição das mesmas para a implementação do sistema proposto. Em seguida no quarto capítulo foram apresentados a proposta do Sistema de Gestão Universitária (SGU), os requisitos que levaram a implementação do Módulo Financeiro e as actividades realizadas. Por fim o quinto e último capítulo a conclusão e recomendações.

² Solis-Cooperativa de Soluções Livres é uma cooperativa de serviços formada no início de 2003, com o apoio da Univates, centro Universitário. A SOLIS implementa soluções tecnológicas livres para os mais variados sectores da academia, indústria, comércio e serviço.

2. Sistema de Informação nas Universidades

Este capítulo contempla a revisão bibliográfica a respeito do tema escolhido, focando os principais assuntos envolvidos, apresentando a perspectiva histórica do sistema de informação e sistemas integrados, as fases da concepção segundo o modelo de RUP (*Rational Unified Process*), e em seguida faz referência ao sistema escolhido para ser customizado e implementado na organização em estudo apresentando as suas principais características.

2.1. Sistemas de Informações

Segundo Silva (2001), um Sistema de Informação (SI) é um conjunto de elementos que interagem de forma interdependente, de modo a formar um todo organizado. Riccio (1989) define um SI como sendo “um conjunto de subsistemas que actuam coordenadamente para, com seu produto, permitirem às organizações o cumprimento de seus objectivos principais.”

Na visão de Carriço (2007) as organizações enfrentam, sem dúvida, desafios consideráveis face às constantes mudanças e evoluções das Tecnologias de Informação (TI), pelas características dos mercados actuais, o que implica que as organizações tenham que tomar decisões num curto espaço de tempo, tanto para as TI que devem adoptar, como também é, necessariamente, para os SI que devem implementar.

2.2. Sistema de Gestão Integrado

Os Sistemas de Gestão Integrados conhecido como ERP (*Enterprise Resource Planning*) tiveram um crescimento acelerado na década de 90 após o seu surgimento. Isto deu-se pelas pressões competitivas, o que levou as empresas a buscarem soluções que lhes permitissem reduzir o custo e diferenciação dos produtos e serviços, forçando-as a reverem os seus processos e a forma de trabalhar. Face a estes acontecimentos, as empresas sentiram a necessidade de coordenar melhor as suas actividades para eliminar desperdícios de recursos, e melhorar o tempo de respostas às mudanças das necessidades do mercado (Sousa, 2000).

Segundo Guedes *apud* Amaral (2006) “uma empresa pode ser visto como um conjunto de processos de negócios que trabalham harmoniosamente integrados visando um obectivo comum...” Os sistema ERP baseam-se em módulos completamente integrados que agrupam funções que atendem um ou mais departamentos.

Para Dávalos & Mülbert (2002) os ERP são sistemas de informações que integram informações e processos entre as diversas áreas funcionais da organização, proporcionando recursos e procedimentos aos utilizadores, para gestão de informações de forma eficiente.

A Uni-CV nesse contexto, também deve usar da mesma ferramenta para apoiar as suas decisões. Para Finger (1997) os processos de Gestão Universitária deveriam ser inovadores e melhorar a interacção entre alunos, docentes, técnicos e, em geral, a comunidade universitária. Reforçando a ideia, (Bernardes & Abreu, 2009) afirma que os colaboradores das Universidades precisam se consciencializar da importância das tecnologias como componente de integração dos diversos departamentos, unidades académicas e administrativas.

Contudo, “a tecnologia de informação volta à Universidade como uma espécie de criatura que desafia o criador a decifrá-la incessantemente” (Marcovith, 1998).

2.3. Fases da Concepção do Sistema de Informação

Segundo (Castro & Moreira, 2007) uma das grandes representantes das metodologias rigorosas é o RUP (*Rational Unified Process*), um recurso de Engenharia de *Software*, criado

pela empresa *Rational Software Corporation* que descreve a maneira de desenvolver um *software* utilizando técnicas comerciais e objectivando o aumento da qualidade do produto criado. É classificado também como um processo de Engenharia de *Software*, que tem como principal objectivo garantir o desenvolvimento de sistemas com qualidade, respeitando os requisitos solicitados pelo cliente, em prazos e custos determinados. Pode ser considerado como um conjunto de experiências e boas práticas adquiridas por diversas pessoas e empresas envolvidas com o desenvolvimento de *softwares*.

Para uma abordagem mais simplificada foi utilizado o modelo de RUP pelo facto de ser totalmente customizáveis, contudo é dito como complexo, sendo recomendável para grandes projectos e equipas de desenvolvimento (Kruchten, 2003).

O Processo Unificado proposto pela RUP foi criado para apoiar o desenvolvimento orientado a objectos, fornecendo uma forma sistemática para se obter vantagens na utilização da Linguagem de Modelagem Unificada (Unified Modelling Language).

A implementação do Sistema de Gestão Universitária baseou-se no modelo de concepção apresentado na figura 1:

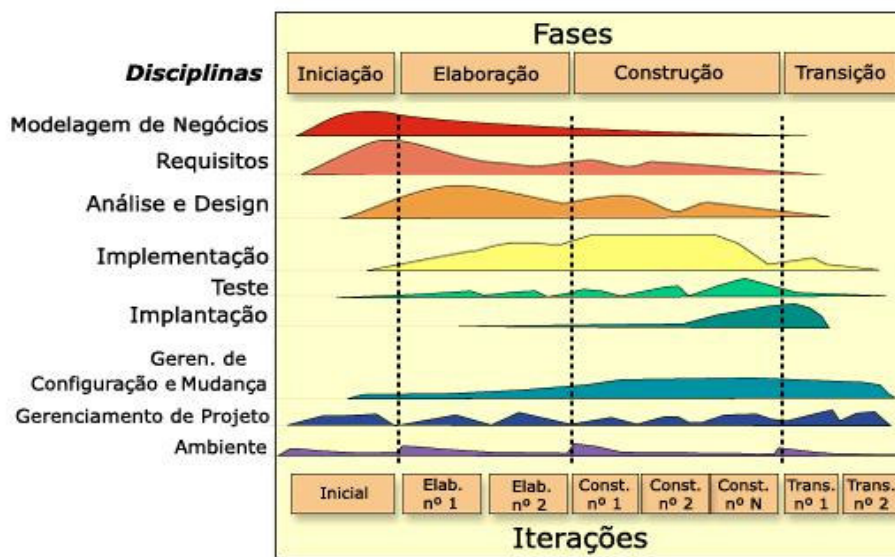


Figura 1: Ciclo do desenvolvimento Utilizando RUP

Fonte: Kruchten (2003)

2.4. Sistema Aberto de Gestão Unificado

Segundo SOLIS (2007), o Sistema Aberto de Gestão Unificado (SAGU) é uma solução criada para auxiliar na gestão das Instituições de Ensino nas esferas académica, administrativa e financeira. O SAGU é constituído em módulos que funcionam em ambientes independentes e oferece aos seus utilizadores um conjunto de ferramentas que integram e optimizam processos e os trabalhos dos diferentes sectores da instituição.

2.4.1. Histórico SAGU

No ano de 1999 o sistema administrativo utilizado pela Univates (Unidade Integrada Vale do Taquari de Ensino Superior), baseado em *softwares* proprietários, não estava a responder de forma satisfatória às demandas. Em Agosto do mesmo ano, Cesar Brod, da Brod Tecnologia e Fábio Wiebbelling, chefe do CPD (Centro de Processamento de Dados) da Univates participaram da *Linux World Conference and Expo* nos Estados Unidos, onde adquiriram algumas experiências de desenvolvimento de aplicações de bases de dados voltadas para a *Web* utilizando a linguagem PHP. Quando retornaram ao Brasil, apresentaram à Reitoria da Univates uma proposta de desenvolvimento de um sistema administrativo, totalmente baseado em *software* livre. A Brod Tecnologia foi então contratada para assumir a gestão dos recursos de informática da instituição. Desde então deu-se o início da criação de um sistema para gestão da Univates. O sistema foi denominado de SAGU (inicialmente um acrónimo para Sistema Aberto de Gestão Universitária, e actualmente Sistema Aberto de Gestão Unificada), (SOLIS, 2009).

A versão do SAGU utilizado para a concepção de uma nova proposta foi a versão 2.0 que apresenta diversas características novas, com suporte a múltiplos temas, reestruturação da camada de geração de recibos bancários com suporte a diversos bancos, reescrita do programa de matrícula com melhorias significativas de *layout*.

2.4.2. Características do SAGU

2.4.2.1. Acesso ao código fonte

O SAGU está sob a licença GPL, ou seja, o seu código fonte é distribuído de forma gratuito. Permite aos técnicos e desenvolvedores de *softwares* desenvolver novas funcionalidades no sistema sem nenhuma dependência para realizar alterações no sistema.

2.4.2.2. Acesso fácil às informações

O SAGU opera com uma interface completamente *web*, moderna e intuitiva, onde todas as informações são disponibilizadas de forma prática e acessível. Para aceder o sistema é preciso um navegador de internet (*Firefox*), o que permite aos utilizadores maior produtividade no desenvolvimento e no cumprimento dos seus serviços.

Um outro aspecto importante de realçar é o desempenho do sistema. As aplicações *web* são cada vez mais conhecidas pelo desempenho oferecido e pela forma de realizar seus processos. O SAGU proporciona rapidez de acesso mesmo em redes de alto tráfego e computadores obsoletos.

2.4.2.3. Segurança

O SAGU permite gerir o perfil de seus utilizadores, limitando o acesso a determinados módulos do sistema e restringindo o uso conforme a política adoptada pela instituição. A gestão dos utilizadores pode ser feita directamente na base de dados ou através de algum serviço de directórios existente na instituição. É possível encriptar todo o tráfego de dados entre o computador e o servidor, permitindo maior segurança na transmissão dos dados na web através da utilização dos protocolos de TLS e SSL.

2.4.2.4. **Licença a custo zero**

O SAGU foi criado para trabalhar com custo zero de licença de *software*. No servidor, é utilizado o sistema operativo Gnu/Linux e a base de dados *PostgreSQL*. A linguagem de programação é PHP, e o *Framework* de desenvolvimento é o Miolo. Todos sob licença GPL, ou seja, de livre distribuição e cópia. O acesso ao sistema pode ser feito através de qualquer sistema operativo, desde que possua um navegador de internet, o que torna o SAGU um sistema multi - plataforma.

2.4.2.5. **Flexibilidade de relatórios**

Os relatórios do SAGU são desenvolvidos com as ferramentas Agata Report pela equipa da SOLIS. Os relatórios podem ser gerados nos formatos PDF e SXW (*OpenOffice*), possibilitando a edição e personalização conforme as necessidades da instituição.

2.4.2.6. **Modularidade**

Por ser um sistema de gestão Académica, o SAGU abrange várias áreas da instituição e, por esta razão, é dividido em módulos onde cada módulo actua no sistema de forma independente dos demais. Esta forma de desenvolvimento possibilita à instituição habilitar apenas os módulos com os quais irá trabalhar.

2.4.3. **Módulos do SAGU**

2.4.3.1. **Módulo Básico**

É o módulo principal do sistema. Neste módulo são registadas informações que são compartilhadas por todos os módulos, como o registo de pessoas (físicas, jurídicas, funcionários), informações geográficas, unidades de ensino, entre outros.

É também neste módulo que residem as configurações do sistema, onde é possível parametrizar o seu funcionamento de acordo com as reais necessidades, como, por exemplo, logótipos, cores, número de registos por página nas listagens, métodos de autenticação, tamanho das janelas, campos dos formulários, formato de data e hora, número de casas decimais, mensagens padrão de inclusão, alteração e exclusão de registos, mensagens de erro e quais módulos estarão disponíveis para uso da instituição.

2.4.3.2. Módulo Académico

Este módulo faz a gestão das informações académicas como cursos, disciplinas, currículos, períodos, matrículas, enfim, tudo o que é necessário para o acompanhamento da vida académica do aluno. O módulo académico possibilita um rigoroso controlo da grade curricular, implementando controlo de versões de curso e disciplina, de requisitos, de liberação de disciplinas, de choque de horários e outros.

O sistema trata das equivalências, conversão de currículo, oferecendo consultas sobre os alunos, quais disciplina já cursou, notas, se é formando, etc. Além disso, gera uma série de documentos, pauta, histórico escolar, livro de matrícula, entre outros.

2.4.3.3. Módulo Financeiro

O módulo financeiro consiste de um sistema de contas a receber. O sistema de contas a receber implementa intercâmbio de arquivos financeiros com bancos, controlo de preços de cursos, incentivos, bolsas, caixas e cheques. Como o SAGU é um sistema multi-empresa e multi-campus, os valores, percentuais de descontos, acréscimos, bolsas, podem ser definidos de forma independente por várias instituições de ensino que compartilhem o mesmo sistema.

O módulo financeiro é consultado pelo módulo académico, quando os dois co-existirem, com o objectivo de integração de informações e assim evitar que um aluno que esteja inadimplimento consiga matricular-se para o próximo período.

Vale realçar que este módulo será abordado mais a frente, tendo em conta que é o módulo principal para elaboração deste projecto.

2.4.3.4. Módulo de Contabilidade

Este módulo possui o controlo contabilístico e centros de custos da instituição e, permite através destes, a integração com os cursos.

2.4.3.5. Módulo Recursos Humanos

Módulo para gerir os salários dos funcionários e professores. Contempla ainda a geração de remessas para folha de pagamento.

2.4.3.6. Módulo Processo Selectivo

O SAGU procura reproduzir os processos de relacionamento de uma instituição de ensino com seus alunos e, no caso de uma instituição de ensino superior, o primeiro contacto com os alunos geralmente ocorre por ocasião de um processo selectivo. Desta forma, este módulo pode ser utilizado para os processos de selecção, como também para concurso de colaboradores.

Este módulo contempla todo o processo de selecção de uma ou mais instituições, desde o cadastro do próprio processo selectivo, às opções, provas, línguas estrangeiras, salas, unidades, locais de prova e candidatos. Além disso, o módulo dispõe de processos internos como divisão dos candidatos nas salas, digitação ou importação das suas notas. As regras de classificação dos candidatos são totalmente customizáveis. O módulo dispõe de diversos relatórios e consultas com os mais diversos dados estatísticos.

2.4.3.7. Módulo Institucional

Através desse módulo é possível registar todos os recursos físicos (salas, laboratórios, auditórios) e materiais (computadores, projectores) da instituição. Com um controlo de versões é possível ter um histórico completo da evolução dos recursos da instituição.

2.4.3.8. Módulo Controlo de Cópias

Pequeno módulo para o controlo da quantidade de fotocópias e impressões tiradas pelo corpo docente e discente,

3. Tecnologias e Ferramentas Utilizadas

Este capítulo insere-se no âmbito das tecnologias e ferramentas utilizadas. Assim sendo, é apresentada de uma forma breve a descrição das mesmas para a implementação do sistema proposto.

3.1. Tecnologias *Open Source*

As soluções *Open Source* estão a ser cada vez mais globalizadas. Actualmente vários organismos adoptam estas soluções nas suas directivas e estratégias. Segundo o artigo publicado pela ASETI (Acessoria de Estratégias de Tecnologias de Informação) Fortaleza, as tecnologias *Open Source* é um programa de computador/*software* com o seu código-fonte disponível ou aberto para toda a comunidade entender, e, caso seja autorizado, pode-se copiá-lo, estudá-lo, modificá-lo e redistribuí-lo sem nenhuma restrição.

Reforçando a ideia, por Yumi (2008) o *software Open Source* é também conhecido como “*free software*” ou *software livre* em que a palavra “*free*” não indica gratuidade, mas sim, liberdade. Afirmando que a liberdade é um dos valores defendidos pelo movimento *Open Source*, além da colaboração e partilha do conhecimento. Devido à ambiguidade do significado do termo “*free software*”, foi criada a expressão “*Open Source*”. De acordo com

(Raymond, 2001), existem quatro liberdades de *softwares livres* sob a licença GPL³ (*General Public License*):

1. “Liberdade de executar o programa, para qualquer propósito”;
2. “Liberdade de estudar como o programa funciona, e adaptá-lo às suas necessidades”;
3. “Liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo”;
4. “Liberdade de aperfeiçoar o programa e liberar os seus aperfeiçoamentos, de modo que toda a comunidade possa beneficiar”.

3.1.1. Vantagens e Desvantagens dos *Softwares Livres*

Os *softwares livres* apresentam as seguintes vantagens:

- Baixo custo, fundamentalmente concentrado em serviços e suporte;
- Menos sujeito a invasões e vírus (maior segurança);
- Continuidade de suporte activa;
- Algumas das alternativas de aplicativos *Open Source* apresentam interfaces para o utilizador final já muito próximas aos seus equivalentes proprietários, facilitando a sua aceitação;
- Maior portabilidade frente às diversas arquitecturas de *Hardware*;
- Maior possibilidade de personalização já incorpora ao aplicativo, com mais alternativas de configurações, possibilitando maior capacidade do aplicativo para atender às necessidades do utilizador;

³ GPL é a designação da licença para *software* livre idealizada por Richard Stallman no final da década de 1980, no âmbito do projecto GNU da *Free Software Foundation* (FSF). Disponível em www.gnu.org, acedido em 2009-10-19

-
- Maior desempenho do sistema, relacionando com uma melhor integração *Hardware/Software*, o que proporciona maior estabilidade dos serviços oferecidos ao utilizador;

Desvantagens:

- Instalação e configuração dos aplicativos ainda tecnicamente complicado pela falta de sistematização e disseminação dos conhecimentos pertinentes;
- Mão-de-obra escassa para suporte e/ou desenvolvimento;
- Resistência cultural do utilizador na migração do aplicativo;
- Interface com o utilizador de alguns aplicativos ainda não totalmente amadurecidos em comparação com os homólogos proprietários;
- Alguma desconfiança por parte dos utilizadores em relação a continuidade e evolução do aplicativo *Open Source*, relacionada com falsa ideia de ausência de comprometimento da equipa envolvida no desenvolvimento do aplicativo;

Com base nos estudos e análises realizados sobre o impacto das tecnologias *Open Source* e *softwares livres*, concluímos que as vantagens apresentadas pelo (Caputi & Pierozzi, 2005) suprem de longe as suas desvantagens.

3.2. Ferramentas Utilizadas

No decurso do presente projecto/estágio tivemos de recorrer as várias tecnologias e ferramentas para proceder o desenvolvimento deste projecto de implementação do Sistema de Gestão Universitária - Módulo Financeiro.

3.2.1. Sistema Operativo Ubuntu

O sistema operativo utilizado para a implementação do SGU foi Ubuntu⁴, um sistema operativo de código fonte aberto e construído em torno do kernel Linux. A comunidade Ubuntu fundou-se nos ideais consagrados na Filosofia Ubuntu, segundo qual as aplicações informáticas devem ser disponibilizadas de forma gratuita, devem ser utilizadas por qualquer pessoa independentemente da sua língua materna ou deficiências e todas as pessoas devem ter a liberdade de alterar e personalizar qualquer aplicação de modo a obterem o que necessitam. Por essas razões são apresentadas algumas características do sistema operativo Ubuntu:

- Baseado em *softwares livres* sob a licença GNU, o que significa que deverá ter a possibilidade de utilizar o *software* da forma que quiser: o código que constitui a aplicação livre deve estar disponível para descarga por qualquer pessoa que o queira alterar, corrigir e utilizar.
- Baseado em Debian, uma das distribuições mais aclamadas, tecnologicamente avançadas e bem suportadas, o Ubuntu ambiciona criar uma distribuição que forneça um sistema Linux actualizado e coerente para computação pessoal e servidores;
- Focando-se na qualidade, o Ubuntu produz um ambiente de trabalho robusto e cheio de características, adequado ao uso quer em casa ou em ambientes comerciais;

3.2.2. *Framework* de Desenvolvimento (Miolo)

Para a criação das páginas dinâmicas foi utilizado o *Framework* MIOLO escrito em PHP utilizando apenas conceitos de POO (Programação Orientada a Objectos) e oferece uma serie de funções para criação de sistemas de informação acessíveis via *web*. Como o MIOLO é o "*kernel*" de todos os sistemas criados, os mesmos podem ser facilmente integrados,

⁴ Publicado por Hardy Heron em Abril de 2008

(Disponível no endereço <https://help.ubuntu.com/8.04/about-ubuntu/C/index.html>, acedido em 2009-10-19).

funcionando como módulos de um sistema mais complexo. Como pré-requisito para utilizar o MIOLO é necessário ter conhecimento de programação com PHP e de POO (Programação Orientado a Objectos). Para criar sistemas utilizando o MIOLO é necessário conhecer algumas regras básicas, entre elas, as definições de separação das classes dos sistemas/módulos, configuração (localização dos arquivos de programas e do MIOLO, BD, temas,), o ciclo de vida de execução de uma requisição do cliente, além das principais classes, métodos e controlos (Gartner & Matos, 2006). O MIOLO adopta uma arquitectura em camadas conforme apresentado na figura 2:

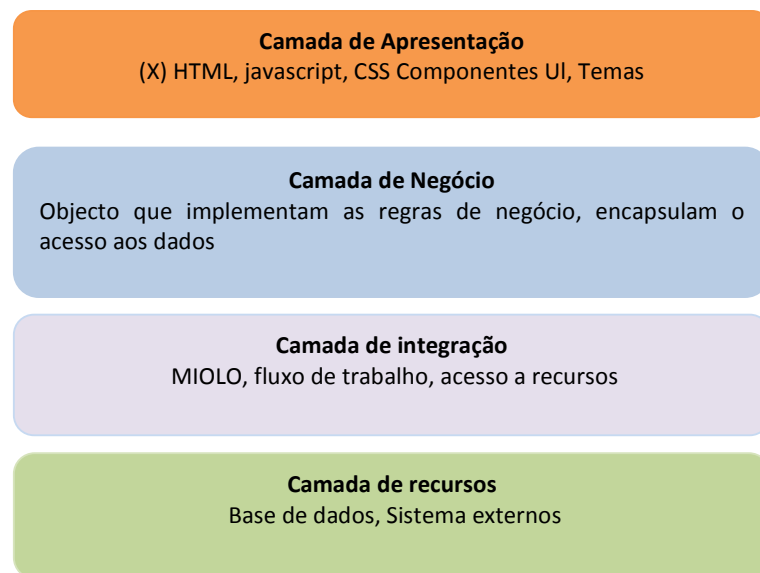


Figura 2: Camadas

Fonte: (Gartner & Matos, 2006)

Na Camada de Apresentação, existe classes do *framework* responsáveis pela geração de arquivos, criado dos controlos HTML e da criação dos *scripts javascript* enviados ao cliente, com base no tema em uso. Engloba também as classes criadas pelos utilizadores para definir a *interface* da aplicação (geralmente nos directórios *forms*, *menus* e *reports* de cada módulo).

Na Camada de Negócio são criadas as classes pelo desenvolvedor para representar o domínio da aplicação (as regras de negócio). São utilizadas pelas camadas UI (*User Interface*) e *handlers*, acedendo a base de dados através da camada Base de Dados.

Na Camadas de Integração são as classes que representam a parte funcional da aplicação, criadas pelo desenvolvedor para fazer o tratamento dos dados enviados pelo cliente. Accede a camada de negócios para desempenhar suas funções e usa a camada UI para definir a saída para o cliente. Estão localizadas no directório *handlers* de cada módulo.

Na Camada de Recurso são as classes do *framework* responsáveis por oferecer recursos e funcionalidades necessárias tanto pelo *framework* quanto pelas aplicações dos utilizadores, encapsulando o acesso a recursos da linguagem.

3.2.3. Servidor *Apache*

Para disponibilizar as páginas dinâmicas⁵ na *web*, é necessário um serviço responsável por todos os recursos disponibilizados, é o servidor *web*. Por exemplo, quando você envia um correio electrónico, efectua compra/pagamentos online através de um formulário, o servidor *web* é responsável por processar todas essas informações. Sendo assim, o servidor *web* é um computador que processa solicitações http, o protocolo padrão da *web*. O servidor *Apache* é o mais bem sucedido servidor *web* livre, extremamente configurável, robusto e de alta performance (Oliveira, 2007).

Segundo estatística da Netcraft, actualmente é o servidor *web* mais utilizado no mundo, com uma quota acima de 60%. É disponibilizado para vários sistemas operativos (*Windows, Novell Netware, OS/2, Linux, etc...*).

⁵ Sites Dinâmico são sites interactivos, permitindo que as informações fluem em ambos os sentidos e em tempo real, tanto no lado cliente como no servidor, (Oliveira, 2007).

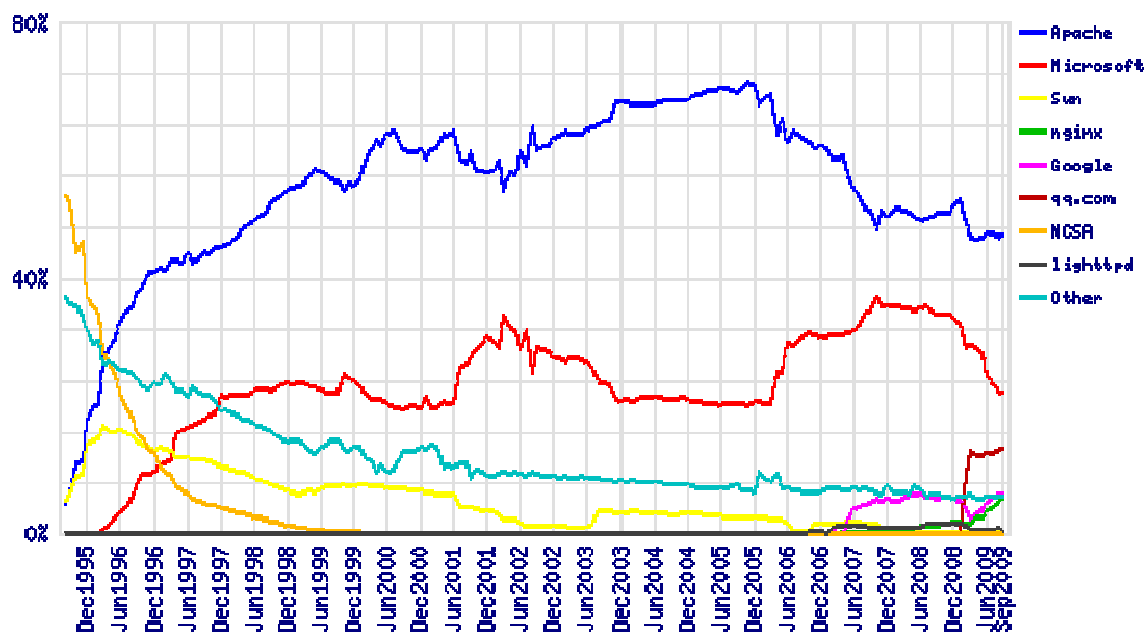


Figura 3: Quotas de mercado de servidores web

Fonte: Netcraft disponível no site: <http://news.netcraft.com>, acedido em 2009

3.2.4. Base de Dados *PostgreSQL*

A base de dados utilizada para implementar o SGU foi o *PostgreSQL* um sistema de código fonte aberto de gestão de base de dados objecto-relacional. O *PostgreSQL* foi escolhido pelas seguintes razões: Base de Dados relacional *Open Source*; altamente escalonável e fácil de administrar, (PostgreSQL, 2009). É projectado para armazenar grandes quantidades de dados e suportar grande número de consulta e actualizações simultânea.

Algumas características apresentada pelo (PostgreSQL, 2009):

- Integridade relacional;
- Suporte a consulta complexa;
- Suporte a gatilhos, *triggers* e procedimentos armazenados no servidor;
- Suporta a tabelas *views*;

3.2.5. Gestão de Base de Dados PostgreSQL: Pgadmin

O *pgadmin* é um projecto de código aberto que permite administrar bases de dados *postgresql*. Segundo o site (www.pgadmin.org acedido em 19-10-2009), *pgadmin* é o mais popular recurso de administração de código fonte aberto e plataforma de desenvolvimento para o *PostgreSQL*. O *pgadmin* é projectado para atender as necessidades de todos os utilizadores, de escrever consultas SQL simples para o desenvolvimento de bases de dados complexos. A interface gráfica suporta todas as características do *PostgreSQL* e facilita a sua administração. É desenvolvido por uma comunidade de especialistas do *PostgreSQL* ao redor do mundo e está disponível em mais de uma dezena de línguas. É um *Software Livre* liberado sob a licença BSD ⁶(*Berkeley Software Distribution*).

3.2.6. Gerador de Relatórios (Agata Report)

Para gerar os relatórios do sistema foi utilizado o *Agata Report*, que segundo (Solis, 2009) é um gerador de relatórios multi-plataforma, uma ferramenta de pesquisas e geração de gráficos que se conecta a vários base de dados, como PostgreSQL, MySQL, SQLite, Oracle, DB2, MS-SQL, Informix, InterBase, Sybase, ou Frontbase e permite exportar os relatórios em formatos como PostScript, plain text, HTML, XML, PDF ou CSV (Excel).

Além das funcionalidades apresentadas no parágrafo anterior, este gerador de relatórios permite ainda criar documentos, como cartas e ligar de forma dinâmica com os dados provenientes do relatório, assim como criar etiquetas de endereçamento e até gerar o diagrama Entidade Relacionamento (ER) completo a partir da sua base de dados.

⁶ BSD permite a redistribuição do software livre sob aviso *copyright*.

3.2.7. PoEdit

PoEdit é um programa utilizado na tradução de software, utilizando o catálogo gettext (Sousa A. , 2008). É leve de instalar, carregar e tem uma interface muito intuitiva, permitindo editar ficheiros com a extensão. PO com a maior das facilidades.

Embora seja um programa simples, é extremamente poderoso na sua função. Utiliza recursos leves e é de fácil utilização. Poedit encontra-se traduzido em 62 línguas diferentes, entre as quais o pt-PT. Poedit funciona em ambientes Windows, Linux, Mac, etc.

3.3. Linguagens Utilizadas

3.3.1. UML (*Unified Modelling Language*)

Para a análise da base de dados foi utilizado a linguagem UML. É uma linguagem para especificação de sistemas, representadas através de diagramas que recorrem a um conjunto simples de símbolos e gráficos. A UML é muito abrangente, porque foi concebida de forma a adequar-se à especificação de qualquer tipo de sistemas (sistemas de informação, sistemas de tempo real, sistemas mecânicos, sistemas geográficos, etc.) e, por outro lado, integra grande parte dos aspectos estruturais e comportamentais que habitualmente caracterizam sistemas (Pedro, 2006).

3.3.2. HTML (*HiperText Markup Language*)

Foi utilizado a linguagem HTML para a publicação de páginas na *web*. Trata-se de uma linguagem não proprietária baseado no SGML (*Standard Generalized Markup Language*), podendo ser criado a partir de uma grande variedade de ferramentas, desde um simples editor de texto até sofisticadas ferramentas de geração de código (Gomes, 2001).

3.3.3. PHP (HiperText Preprocessor)

Para (Oliveira, 2007), PHP é uma linguagem de *scripting* do tipo *server-side*, mas com característica de ser *open source*. O PHP pode ser utilizado na maioria dos sistemas operativos, incluindo *Linux*, diversas variantes *UNIX*, *Microsoft Windows*, *Mac OS X* e *RISC OS*. A linguagem PHP proporciona o desenvolvimento de aplicações para base de dados utilizando uma interface *web*. Esta linguagem foi criada por Rasmus Lerdorf e actualmente é a linguagem que mais cresce na Internet. Esta linguagem suporta um vasto leque de bases de dados. As páginas PHP têm extensão **.php**⁷.

⁷ Para mais informação, poderá consultar o site <http://www.php.net>

4. Implementação do Módulo Financeiro

Neste serão apresentados a proposta do Sistema de Gestão Universitária (SGU), os requisitos que levaram a implementação do Módulo Financeiro, os casos de uso e as actividades realizadas para a implementação do Módulo Financeiro.

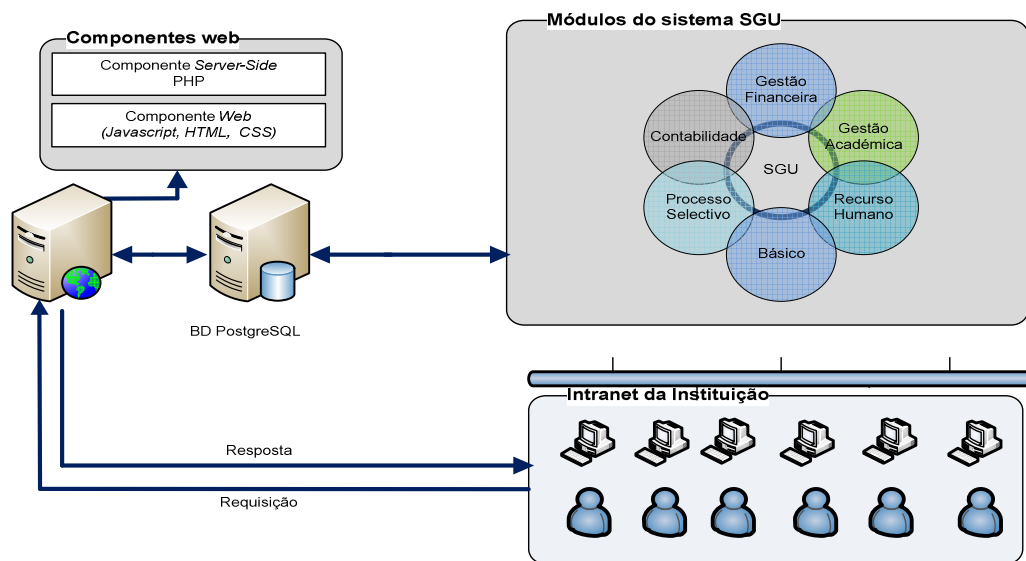
4.1. Proposta do Sistema de Gestão Universitária

Com a crescente pressão para que as Universidades racionalizem os seus processos de gestão, várias delas vêm na integração dos Sistemas de Gestão uma excelente oportunidade para automatização dos processos para uma gestão centralizada segundo Chaib *apud* Godini (2005). A proposta de um Sistema de Gestão Universitária denominado de (SGU) revelou-se um caminho adequado para a gestão da informação, com o objectivo de otimizar a integração da informação. O SGU visa integrar todos os processos de negócio da Universidade de Cabo Verde em busca de excelência para uma gestão participativa e integrada. O sistema proposto herda algumas características que o SAGU apresenta, código fonte aberto em caso de futuras customizações, acesso fácil às informações dentro de um ambiente web, segurança, flexibilidade de relatórios e modularidade. Os módulos que constituem o SGU são: módulo Básico, Módulo Académico, Módulo Financeiro, Módulo Contabilidade e Módulo Recursos Humanos.

4.1.1. Arquitectura do SGU

Segundo (Pereira, 2008), a independências das aplicações de cliente, relativamente à localização física dos dados é preponderante para o rápido funcionamento e garantia de maior das aplicações. Uma das funções mais importante dentro de uma infra-estrutura tecnológica é o acesso aos dados. O SGU é arquitectado utilizando uma arquitectura cliente-servidor, um modelo de computação que visa tirar partido dos sistemas distribuídos (Rafael, 1999). Os módulos que constituem o sistema proposto funcionam de forma independente, excepto o módulo básico que é o módulo principal do sistema, onde serão feitas todas as configurações e o registo dos processos da instituição. O Módulo Financeiro foi implementado em conjunto com os módulos (Académico e Básico) com o objectivo de controlar o processo Financeiro do aluno com a Universidade.

A figura seguida apresenta uma arquitectura Cliente/Servidor do tipo *server-side*, o conteúdo é gerado de forma dinâmica no servidor *web*, na sequência de *scripts*. Actualmente, as aplicações *web* utilizam cada vez mais soluções *server-side*, por apresentarem algumas vantagens relativamente à solução *client-side*⁸ (Oliveira, 2007).



⁸ *client-side*: as aplicações do tipo *client-side* são executadas do lado do cliente no browser ou no sistema operativo, isto é qualquer elemento de programação que seja executado ao lado do cliente é completamente transparente para o servidor (Oliveira, 2007).

Figura 4: Arquitectura de funcionamento de SGU

4.2. Razões da escolha do SAGU

Segundo o Director dos Serviços Técnicos da Uni-CV (Celestino Barros), entre os *softwares* livres analisados para a Gestão da Universidade de Cabo Verde foi apontado o SAGU como a melhor solução para a Implementação de um Sistema de Gestão Universitária. Tendo em conta que é um sistema baseado em tecnologias *Open Source*, capaz de responder às necessidades de automatização das áreas Administrativas da Universidade.

Com o objectivo de identificar os sistemas que melhor se adequa à Uni-CV foi feito um estudo comparativo entre seis *softwares*, sendo dois deles livres conforme se descreve na página seguinte:

Características	Sistemas					
	SAGU	TesEscola	AdX	UniMestre	SophiA	Corpore RM Education al
Por Categoria de Instituição de Ensino						
Ensino Básico	X	X		X	X	X
Escola Secundário	X	X		X	X	X
Ensino Superior	X	X	X	X	X	X
Por características funcionais						
Controlo Académico	X	X	X	X	X	X
Controlo Financeiro	X	X	X	X	X	X
Controlo de Processo Selectivo	X		X	X		
Recursos Humanos	X	X	X		X	X
Biblioteca	X	X	X	X	X	X
Por características operacionais						
Plataforma	Web	Desktop/ Web	Web	Desktop/ Web	Desktop/ Web	Desktop/ Web
Distribuição	Livre	Proprietário	Livre	Proprietário	Proprietário	Proprietário

Tabela 1: Softwares analisados

Fonte: Equipa de Desenvolvimento do Sistema de Gestão Universitário (Serviços Técnicos)

Para analisar esses *softwares* foram estudadas as suas principais características:

- Tipo de escola que o *software* visa atender (Ensino Básico, Secundário ou Superior);
- Possuir controlo académico (notas, faltas, e controlo de outras informações académicas de alunos e professores), controlo Financeiro, controlo de Processo Selectivo, Gestão de Recursos Humanos, Gestão de uso de Recursos (salas e equipamentos);
- Plataforma em que foi implementado e a forma de distribuição.

Nota-se que a maioria dos *softwares* apresentados na tabela acima, foram completamente implementados ou ao menos contêm módulos implementados para acesso via *web*. A maioria tem carácter diversificado, atendendo às necessidades de Escolas de Ensino Básico, Secundário e Superior. Todos os *softwares* possuem controlo académico (gestão de notas, faltas, e outras informações de alunos e professores), gestão de biblioteca e gestão financeiro.

Algumas tarefas administrativas como a Gestão de Recursos Humanos e Controlo de Processo Selectivo não têm suporte em todos os *softwares*, o que não acontece com o SAGU.

Face às características apresentadas pelo sistema adoptado (SAGU), deu-se a criação de um novo Sistema para Gestão Universitária (SGU). Desde então foram feitas diversas modificações para que o sistema proposto se adapte da melhor forma à realidade da Universidade de Cabo Verde.

4.3. Implementação do Módulo Financeiro

O Módulo Financeiro é integrado com o ambiente académico, tendo grande parte das movimentações e acções realizadas, integradas com este ambiente. Destacamos as principais funcionalidades do módulo financeiro:

- Gerar mensalidades no processo de matrícula, dividida em números de parcelas;
- Actualização dos títulos a receber;
- Cancelamento de títulos sincronizado a movimentações de matrículas;
- Geração de títulos relacionados a extras curriculares;
- Controlo de caixas, indicando as contas internas ou bancárias da instituição
- Transferências entre contas e sincronizações bancárias;
- Visualização de relatórios de gestão;
- Controlo de fluxo de caixa;

4.4. Actividades Planeadas

No âmbito deste projecto foi efectuado um estudo das funcionalidades possíveis a serem implementadas. Salvaguarda-se que é importante nesta fase fazer um rigoroso estudo conceptual da aplicação, identificando os seus requisitos, funcionalidades e casos de uso. Esta análise permitiu-nos compreender o modo de funcionamento do sistema, facilitando a sua implementação.

Para a implementação e customização do Módulo Financeiro foram identificadas as seguintes fases:

1. Estudo das ferramentas e linguagens utilizadas.
2. Análise das funcionalidades que se pretendem implementar;
3. Análise de base de dados;
4. Realizações de testes e correcção de possíveis erros no sistema;

A página seguinte apresenta o cronograma das actividades planeadas para a implementação do Módulo Financeiro:

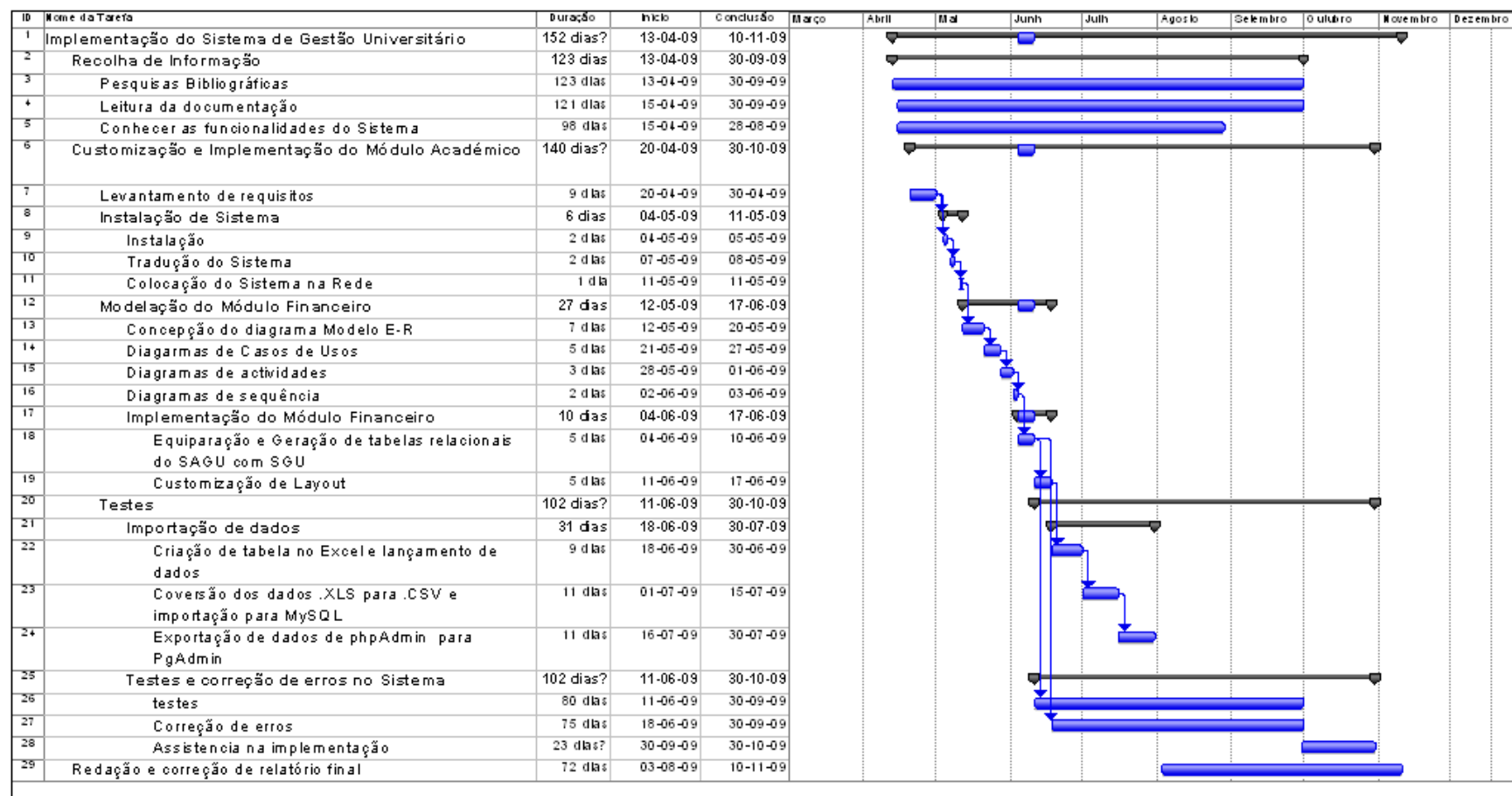


Figura 5: Cronograma das actividades

4.5. Dificuldades Sentidas Durante o Estágio

No início da implementação do

4.6. Descrição dos Requisitos

4.6.1. Levantamento dos Dados

Escolhemos a técnica de entrevista não estruturada como técnica de levantamento de dados sobre a forma como se processa a cobrança de propinas. A entrevista foi realizada directamente com um dos responsáveis pelo serviço financeiro da Uni-CV Campus Palmarejo.

4.7. Descrição da Situação Actual

No processo actual, para que o aluno possa efectuar o pagamento de propina, o funcionário responsável pela cobrança preenche um formulário (caderneta), no qual emite um recibo de cobrança. Por sua vez o funcionário encaminha o formulário de cobrança realizado no final do dia para o responsável do sector financeiro que em seguida faz o lançamento dos dados numa folha de Excel. O processo é feito de forma manual desde sua origem até à sua conclusão. Todo fluxo financeiro é controlado através de um formulário e salvaguardado numa folha de *Excel*, gerido pelo funcionário responsável do sector financeiro.

4.7.1. Análise de Requisitos

Com a realização da entrevista foram encontrados os seguintes problemas no sistema actual:

- O processo de cobrança é feito manualmente;
- Controlo de pagamento manualmente;
- Dificuldade na limitação de fraude;

-
- Tempo de execução das tarefas de cobrança não responde eficazmente à demanda;
 - Falta de segurança no armazenamento de documentos e informações;
 - Dificuldade na busca de determinadas informações;

4.7.2. Solução Proposta

De acordo com o contexto apresentado a solução proposta para a implementação do sistema visando a otimização do processo, apresenta as seguintes características:

- Controlo de cobrança de propinas;
- Relatórios de cobrança (mensalidades, pagamentos, etc);
- Registo dos títulos de cobrança e previsões de lançamentos;
- Actualização de Títulos e Lançamentos;
- Maior segurança dos dados;
- Maior acessibilidade das solicitações;
- Rapidez na busca de determinadas informações;

4.8. Requisitos de *Hardware* e *Software*

Como pré-requisito para a implementação do sistema, necessitou-se que suas dependências sejam instaladas e configuradas. A aplicação foi implementada com as seguintes tecnologias e ferramentas.

- Sistema Operativo Ubuntu 8. 04 (Disponível no www.ubuntu.com);
- Servidor *Apache* 2.2 (Disponível no www.apache.org);
- Base de Dados *PostgreSQL* (Disponível no www.postgresql.org);
- *Agata Report*;
- Linguagem HTML, PHP;

Também vale realçar que as tecnologias e ferramentas utilizadas para a implementação do SGU (Módulo Financeiro) foram descritas no capítulo 2.

4.8.1. Requisitos de Instalação

Os requisitos mínimos de *hardware* utilizados para a instalação do Sistema de Gestão Universitária:

- Processador Pentium IV ou Similar;
- Memória RAM de 512 MB (1GB recomendado);
- HD de 80 Gb (SATA recomendado);

Requisitos utilizados para ambiente de produção:

- Processador Intel Pentium (R) Dual Core CPU 1.80GHZ;
- Memória RAM 1GB;
- Disco Rígido 160GB;

4.9. Projecto Conceptual

Com base nas análises de requisitos e a solução proposta, procedeu-se a definição dos casos de usos do módulo conforme define a especificação da UML utilizando para a ferramenta Visual Paradigm.

A figura 6 mostra a solução proposta através do diagrama de casos de uso, onde o mesmo determina quais são os actores envolvidos no sistema (pessoas que interagem com o sistema) e as funcionalidades que tem acesso.

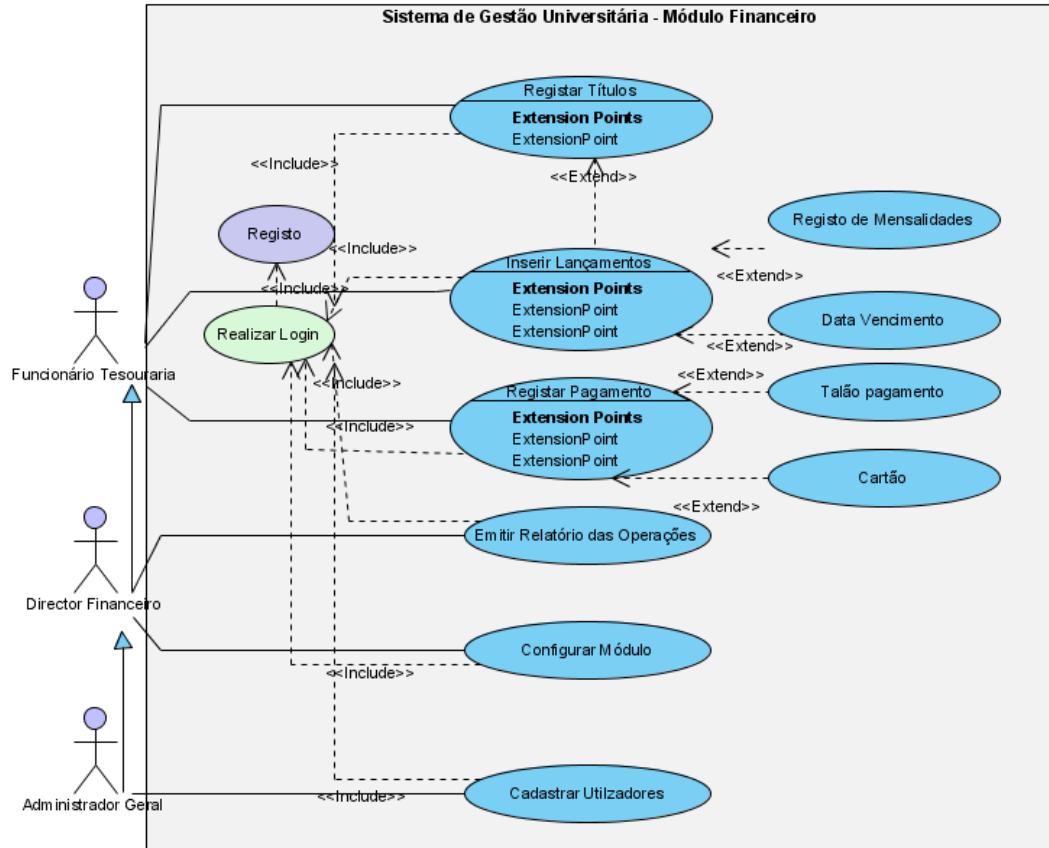


Figura 6: Caso de uso para processamento de Títulos a receber

4.9.1. Descrição dos Casos de Utilização

Para melhor organizar e visualizar o caso de uso criou-se uma tabela que contém as informações sobre os casos e uma referência aos requisitos que compõem o caso de uso. Na tabela é apresentada a descrição dos casos de uso que compõem o Módulo Financeiro para o processamento de títulos a receber.

As tabelas seguidas apresentam as descrições dos actores e casos de utilização de sistema:

Descrição dos actores do sistema:

Actores	Descrição
Funcionário Tesouraria	Pessoa responsável pela cobrança de propinas e registo das operações (mensalidades, pagamento).
Director Financeiro	Pessoa responsável pelo funcionamento do Módulo Financeiro, altera data de vencimento, emite relatórios das operações e define políticas do uso.
Administrador do Sistema	Pessoa responsável pela gestão do módulo, customização e implementação de políticas e faz manutenção do sistema.

Tabela 2: Descrição dos Actores

Descrição do caso de utilização:

Caso	Actor	Descrição
Utilização		
Registar Títulos	Funcionário Tesouraria	Faz o registo os títulos
Inserir Lançamentos	Funcionário Tesouraria	Inserir informações de lançamentos
Registar Pagamentos	Funcionário Tesouraria	Faz o registo de pagamentos para cada prestação (pode ser Integral ou parcial)
Emitir Relatórios das Operações	Director Financeiro	Emite relatórios das operações (mensalidade, pagamentos, inadimplementos, etc.)
Configurar Módulo	Director Financeiro	Define parâmetros de configuração, altera layout, define política do uso.
Registar utilizadores	Administrador Geral	Acessos a todas as funcionalidades do módulo além de fazer o registo de utilizadores no sistema

Tabela 3: Descrição de Caso de Utilização

4.9.1.1. Caso de Uso - Efectuar Login

Actores: Funcionário / Administrador.

Descrição: Este procedimento consiste em realizar a operação de autenticação dos utilizadores no sistema. O utilizador (Funcionário ou Administrador) fornece seu nome de utilizador (*username*) e a respectiva palavra-chave (*password*) de acesso ao sistema e é validado através da base de dados do sistema.

Pré-condições: É necessário que o utilizador esteja registado no sistema.

Pós-condições: A sessão é iniciada.

Caso normal: O utilizador preencherá as suas credenciais na página de autenticação. Estas credenciais são compostas pelo nome do utilizador e pela palavra-chave. A verificação do nome do utilizador é *case sensitive*. Caso correctamente inserido, o utilizador poderá após fazer o *login* entrar no modo de visão completa do Módulo Financeiro, podendo ver todas as suas funcionalidades.

Caso alternativo: Se o utilizador tiver introduzido incorrectamente as suas credenciais, será mostrado uma mensagem (Erro na identificação do utilizador!) e número restantes de tentativas na mesma página de *login*.

4.10. Projecto Lógico

Para o melhor entendimento do sistema foi feito um estudo da base de dados do sistema para se melhor adapta-lo segundo os requisitos proposto.

A figura 7 mostra as tabelas que compõem a base de dados, seus atributos e os respectivos relacionamentos. Realçarmos que as descrições das tabelas que constituem a base de dados foram apresentadas no anexo.

4.11. Diagrama de Actividade

4.11.1. Processo Criação de Títulos a Receber

Para a efectuação do processo de cobrança de propinas, são gerados títulos a receber com base nas definições do ano lectivo em que aluno se encontra matriculado. Títulos a receber consiste no número de parcelas a ser registado para a cobrança de propina.

A figura 8 representa o processamento de títulos a receber segundo o caso de uso apresentado anteriormente:

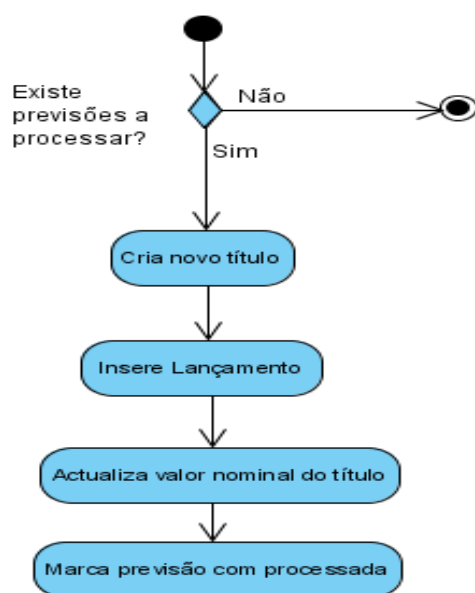


Figura8: Diagrama de actividade para criar títulos a receber

Para melhor se entender o processo de criação de títulos a receber é apresentado de seguida o diagrama de sequência:

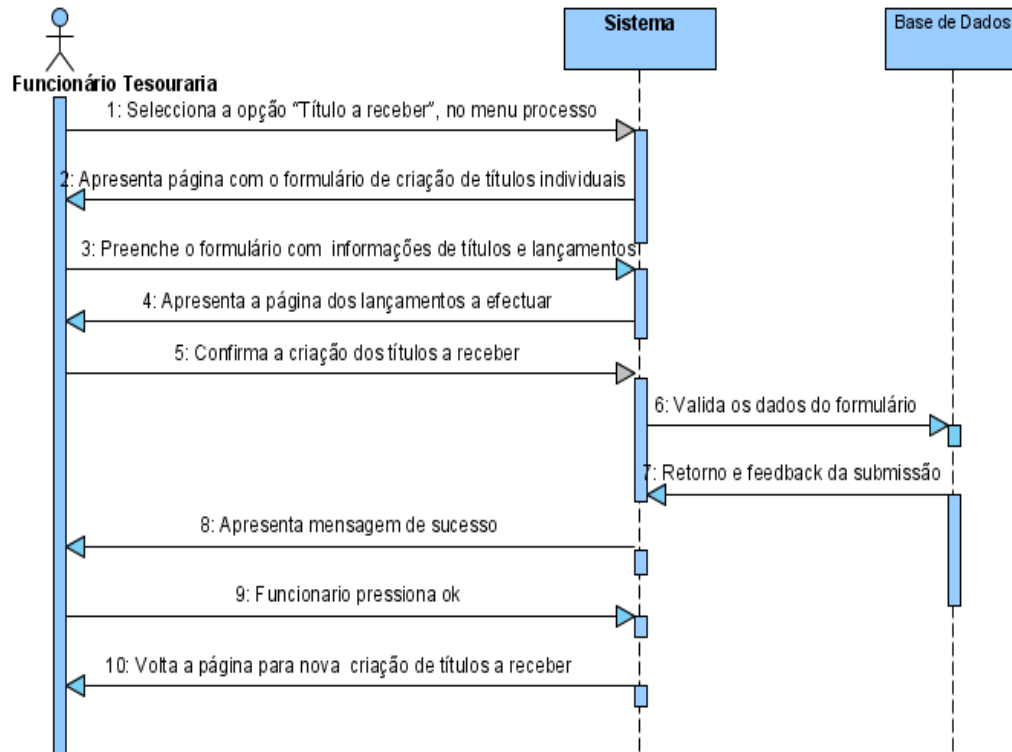


Figura 9: Digrama de sequência para criar títulos a receber

4.11.2. Processo para a Pagamento de Propinas

Após de gerar títulos a pagar o funcionário tesouraria faz o registo da cobrança com base nos títulos a receber (mensalidades).

A figura 10 representa o processamento para o pagamento de propinas:

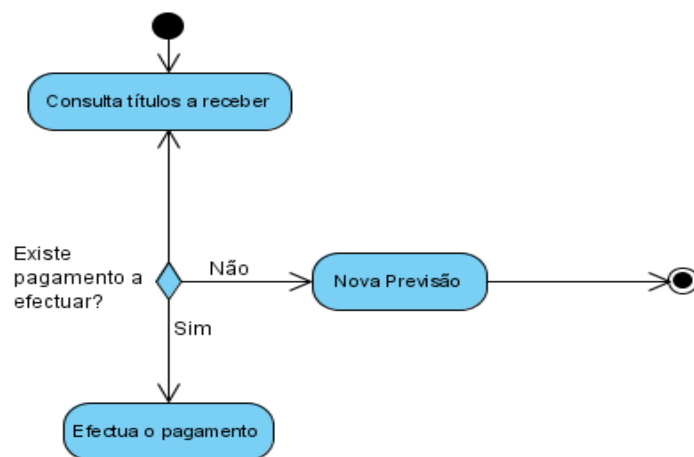


Figura 10: Diagrama de actividade para pagamento de propinas

Em seguida apresenta-se o diagrama de sequência para o registo do pagamento de propina.

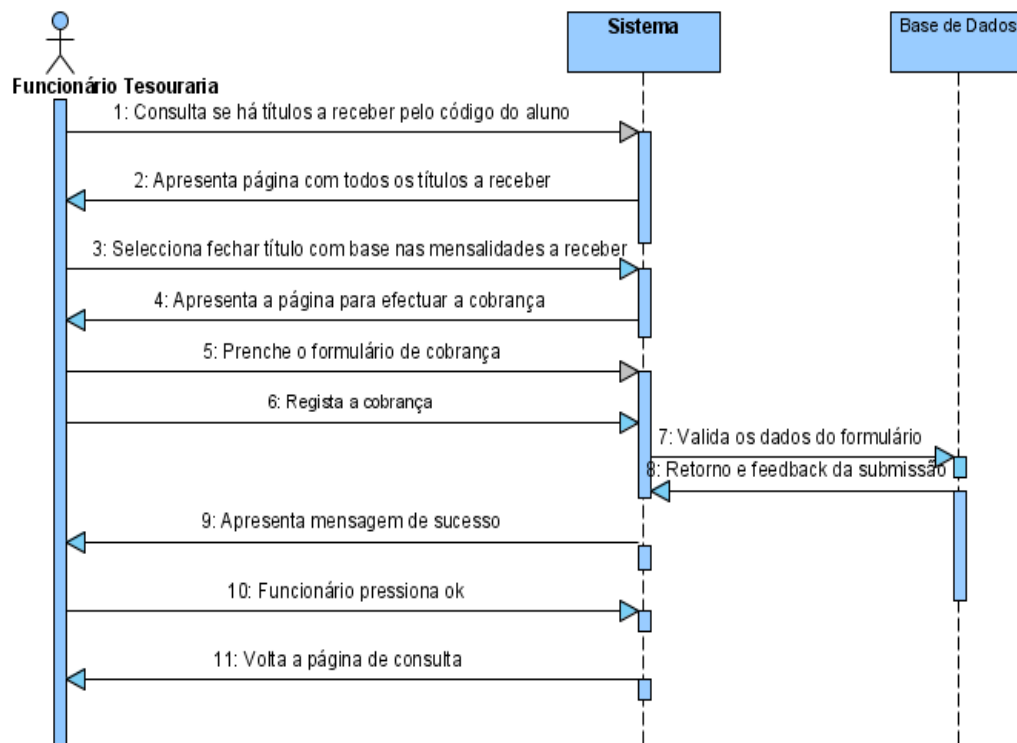


Figura 11: Diagrama de sequência para pagamento de propina

4.12. Testes

Para a concepção deste projecto não foi utilizado nenhum método específico para realização dos testes. No decorrer das customizações planeadas foram realizados testes utilizando registos reais dos documentos no sistema em colaboração com o pessoal dos serviços financeiros e a equipa de desenvolvimento do sistema, para inserção dos dados, segurança, apresentação.

Na fase da inserção dos dados destacamos a inserção de campos em branco, preenchimentos de espaço e tamanho máximo e mínimo dos dados inseridos no sistema de forma a verificar a consistência dos dados.

Relativamente à apresentação, foram testados basicamente a usabilidade do sistema perante diversos utilizadores. Estes tipos de testes permitem verificar quais as funcionalidades onde os utilizadores mostraram mais dificuldade na sua utilização, revelando os diversos aspectos para possíveis melhorias.

Outro aspecto importante é a compatibilidade dos *browsers*. Uma das características principal do sistema é que pode ser utilizada em diversas plataformas. A fim de evitar incompatibilidade dos *browsers* foram testados, o Firefox, Internet Explorer e Opera. O Firefox demonstrou melhor compatível no processo de testes realizados.

A nível de segurança foram definidos e testados os privilégios para cada tipo de utilizadores na utilização do sistema. Os utilizadores somente terão disponíveis a *interface* necessária para a execução das suas tarefas.

4.13. Apresentação da Implementação

Para concluir o processo de customização e implementação do Módulo Financeiro para a gestão e controlo de propinas, são apresentadas algumas *interfaces* gráficas do sistema, projectado de forma a representar o resultado do trabalho exercido.

O SGU permite gerir os utilizadores através do serviço LDAP⁹. A cada utilizador dentro de rede é atribuído um nome de utilizador (*username*) e a palavra-chave (*password*) para o acesso. A primeira tela representa a página de acesso aos módulos do Sistema de Gestão Universitária.

Conforme referida no caso de uso a figura 12 apresenta a página de autenticação dos utilizadores:

⁹ LDAP é uma tecnologia em rápido crescimento de acesso à informação contida em directórios para gestão das permissões. Uma vez que é um standard aberto, independente da implementação, o LDAP fornece uma arquitectura extensível para o armazenamento centralizado e gestão de informação que necessite de estar disponível para os sistemas e serviços distribuídos dos dias de hoje.

Disponível em http://marco.uminho.pt/~osg/CC-I-LESI/ficha5/LDAP_pt.pdf Acedido em 08 de Setembro de 2009.

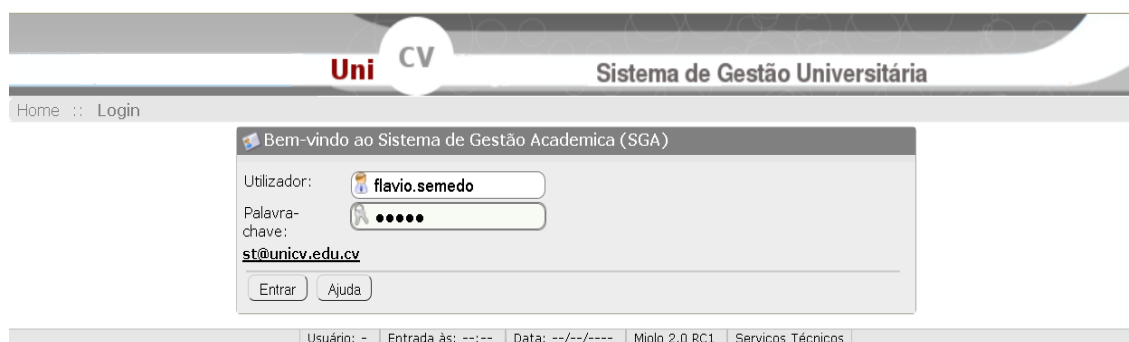


Figura 12: Página de acesso do SGU

Após da autenticação, o utilizador entrará no modo da visão completa do Módulo Financeiro. Nesta página o utilizador terá acesso a todas as funcionalidades que se deseja aceder.

A figura 13 apresenta a página principal para o acesso ao Módulo Financeiro:



Figura 13: Página principal

A figura 14 representa o formulário dos dados referente ao registo dos títulos a receber para cada aluno.

Uni CV Sistema de Gestão Universitária

Uni- CV Favoritos Sistema

Home :: Financeiro :: Processo :: Título a receber :: Geração individual de títulos

Geração de título

Principais informações do título

Pessoa:*

Curso:* Versão

Origem:*

Plano de contas:*

Centro de custo:*

Unidade: --Selecione--

Política:*

Débito automático:* ☐ Sim ☒ Não

Data de emissão:* 17/12/2009

Conta bancária:* 6 2326452-01: UNICV

Sector: --Selecione--

Comentários:

Figura 14: Criar Títulos

Após à criação de títulos a receber, são inseridas as informações dos lançamentos que consiste no valor pago para cada prestação e a respectiva data de vencimento.

A figura 15 apresenta o formulário das informações do lançamento:

Informações do lançamento

Número de parcelas:*

Lançamentos por parcela:* --Selecione--

Valor sugerido:*

Data de vencimento sugerida:*

Código da operação sugerida*

Próximo

Usuário: flavio Entrada às: 15:14 (00:00) Data: 17/12/2009 Miolo 2.0 RC1 Serviços Técnicos

Figura 15: Informações do Lançamento

Para o registo da cobrança de propinas, o funcionário tesouraria com base nos títulos gerados para cada aluno poderá efectivar o pagamento da propina:

A figura 16 apresenta a página do registo para a liquidação do título receber:

Uni CV Sistema de Gestão Universitária

Uni-CV Favoritos Sistema

Home :: Financeiro :: Processo :: Fechar título

Fechar título

Código do título: 4403

Tipo de fechamento: Integral

Operador:*

Caixa:*

Espécie:*

Valor nominal: 9000.00

Valor: 10710.00

Operação (somente crédito):

Comentários:

Fechar título

Usuário: flavio Entrada às: 15:14 (00:04) Data: 17/12/2009 Miolo 2.0 RC1 Serviços Técnicos

Figura 16: Fechar Título

As demais *interfaces* gráficas do trabalho não foram apresentados pelo facto do trabalho não se tornar extenso.

5. Conclusão e Recomendações

A realização do estágio final de uma licenciatura, assim como a apresentação do respectivo relatório, afigura-se como uma das tarefas indispensável, tendo em conta a experiência obtida através da implementação do SGU “Módulo Financeiro” baseado em *softwares livres* na Uni-CV. A adopção das soluções livres na Universidade, como parte principal do projecto de implementação vem gerando resultados positivos, porque foram atingidos, na generalidade, os principais objectivos traçados pela Universidade de Cabo Verde. Para atingir o objectivo principal deste trabalho, baseou-se em objectivos secundários: estudar as necessidades actuais da área financeira da Universidade, bem como procurar soluções capazes de gerir os serviços financeiros e estudar as tecnologias e as ferramentas para a sua implementação e ainda definir o protótipo, testes e verificações de erros.

Para que os colaboradores conheçam a realidade da instituição, entender os factores que a determinam e controlar a sua evolução, tomando decisões acertadas, há que dispor de informações necessárias, adequadas, actualizadas e processadas de forma integrada. Por esta razão, a Implementação do SGU “Módulo Financeiro” revelou um caminho adequado para que estas decisões sejam disponibilizadas de forma clara e precisa.

Importa afirmar que este trabalho proporcionou-nos a aquisição de novas experiências a nível de implementação e sistemas de informação. Ajudou-nos a pôr em prática grande parte dos conhecimentos adquiridos ao longo do curso. Para além dessa evolução evidente a nível pessoal, não se pode deixar de referir o facto de contribuir bastante para a instituição, pois era esse o seu fim. Tratou-se de um trabalho essencialmente prático, o que o tornou a realização deste projecto mais relevante e precisa. À medida que as customizações eram feitas, o resultado era imediatamente visível.

Este projecto é o reflexo da dedicação e esforço empregado desde do início até ao final do curso de Licenciatura em Tecnologias de Informação e Comunicação, pois, tudo o que foi aprendido no decorrer do curso foi utilizado para a concepção deste projecto. A experiência adquirida durante esta fase abrirá grandes oportunidades no mercado do trabalho, especialmente na área de desenvolvimento baseado em *softwares livres*.

Mesmo que tenhamos conseguido atingir o principal objectivo deste projecto, que é o de implementar um Sistema de Gestão Universitária “Módulo Financeiro”, fica como sugestão os seguintes pontos para desenvolvimento futuro:

- Ampliar o sistema para a disponibilização de consultas e pagamentos de propinas online;
- Desenvolver um sistema de alerta de pagamento propina via dispositivo móvel;
- Disponibilizar relatórios financeiros online aos gestores da instituição;
- Proporcionar o desenvolvimento de novos módulos para aplicação no sistema de SGU, permitindo a sua utilização nas mais diversas aplicações;
- Integrar o Sistema de Biblioteca, facilitando a interacção dos sectores bibliotecário, serviços financeiros, académicos, a fim de proporcionar mais agilidade no acesso às informações dos alunos que utilizam recursos disponíveis na biblioteca.

Referências Bibliográficas

Alsene, E. (1999). The computer integration of the enterprise.

ASETI. (21 de Setembro de 2007). *Directrizes para Política de Software Livre do Governo do Estado de Ceará*. Obtido em 25 de Setembro de 2009, de <http://softwarelivre.ceara.gov.br/categoria1/documentos-oficiais-arqs/Relatorio-Final%20-Diretrizes-de-SL.pdf>

Attila, E. B. (2004). *Proposta de Sistematica para Orientar a Implantação de Softwares Integrados de Gestão*. Obtido em 19 de Novembro de 2009, de <http://www.lume.ufrgs.br/bitstream/handle/10183/4720/000459175.pdf?sequence=1>

Bernardes, J. F., & Abreu, A. F. (2009). A contribuição dos sistemas de informações na gestão universitária.

Caputi, E., & Pierozzi, I. J. (Dezembro de 2005). Obtido em 25 de Setembro de 2009, de http://www.cnpm.embrapa.br/publica/download/doc41_opensource.pdf

Carriço, N. R. (2007). *Arquitectura da Função de Sistemas de*. Obtido em 01 de Agosto de 2009, de https://repositorio.utad.pt/bitstream/10348/47/1/msc_nrfcarriço.pdf

Castro, I. C., & Moreira, A. M. (2007). *METODOLOGIAS DE DESENVOLVIMENTO: UM COMPARATIVO ENTRE EXTREME PROGRAMMING E RATIONAL UNIFIED PROCESS*. Obtido em 17 de Novembro de 2009, de <http://www.frb.br/ciente/BSI/BSI.CASTRO.%20et%20al%20.F2%20.pdf>

Chaib, E. B. (Março de 2005). *Proposta para a Implementação de Sistema de Gestão Integrado de Meio Ambiente*. Obtido em 17 de Junho de 2009, de <http://www.ppe.ufrj.br/ppe/production/tesis/ebdchaib.pdf>

Dávalos, R. V., & Mülbet, A. L. (Outubro de 2002). *Implantação de um Sistema Integrado de Gestão*. Obtido em 19 de Novembro de 2009, de http://www.abepro.org.br/biblioteca/ENEGEP2002_TR90_0239.pdf

Finger, A. P. (1997). *Liderança e administração na universidade*. Florianópolis: UFSC/CPGA.

Gartner, V. C., & Matos, E. E. (16 de Maio de 2006). *Framework Miolo 2.0*. Obtido em 17 de Novembro de 2009, de http://www.miolo.org.br/files/files/site_miolo///miolo2_overview.pdf

Gomes, D. C. (Julho de 2001). *Sistema de Recolha de Documentos na Web*. Obtido em 18 de Novembro de 2009, de <http://xldb.fc.ul.pt/xldb/publications/RelFinalTarantula43Alt.pdf>

Guedes, A. L., & Amaral, M. C. (2006). *ERP-Enterprise Resource Planning*. Obtido em 21 de Agosto de 2009, de http://andre.guedes.googlepages.com/ERP_Enterprise_Resource_Planning.pdf

Kruchten, P. (2003). *Introdução a RUP*. Obtido em 17 de Novembro de 2009, de http://www.personati.com/artigos/Pu_RUP.doc

KRUCHTEN, P. (2003). *Introdução ao RUP - Rational Unified Process*. Nacional: Ciencia Moderna.

KRUCHTEN, P. (2003). *Introdução ao RUP - Rational Unified Process*. Obtido em 17 de Novembro de 2009

Marcovith, J. (1998). *A universidade (im)possível*. São Paulo: Futura.

Oliveira, H. (2007). *Dreamweaver CS3 e 8*. Lisboa: FCA.

Pedro, N. R. (2006). *Desenhar Bases de dados com UML*. Cidade de Manchester: Lisboa.

pgAdmin. (2009). *PostgreSQL Tools*. Obtido em 09 de Setembro de 2009, de www.pgadmin.org

PostgreSQL. (2009). *PostgreSQL Global Development Group*. Obtido em 14 de Agosto de 2009, de <http://www.postgresql.org/files/documentation/pdf/8.4/postgresql-8.4.0-A4.pdf>

Rafael, G. M. (Setembro de 1999). *Sistema de Informação Baseado numa Intranet*. Obtido em 01 de Outubro de 2009, de <http://www.esev.ipv.pt/tear/Recursos/17/Dissertacao-Rafael.PDF>

Raymond, E. (2001). *Musings on Linux and open source by an accidental revolutionary*. Sebastopol, CA: O'Reilly.

Riccio, E. (1989). *Uma Contribuição ao Estudo da Contabilidade como Sistema de*. São Paulo: FEA/USP, 1998.

Rodolfo, M. (2007). *Implementação de Sistemas ERP*. Obtido em 19 de Novembro de 2009, de http://sapr3abap.files.wordpress.com/2008/02/monografia_erp.pdf

Silva, R. O. (2001). *Teoria de administração*. São Paulo: Pioneira Thompson Learning.

Solis. (2009). *Agata Report*. Obtido em 13 de Novembro de 2009, de http://www.agata.org.br/handler.php?module=sites&action=view§ion=23&sitesdbname=fred_agata

SOLIS. (2009). *Manual SAGU*. Obtido em Setembro de 27 de 2009, de http://rumix.com.br/ensino/documentos/aoo/AOO_SEI.pdf

Sousa, C. A. (Maio de 2000). *Sistemas Integrados de Gestão Empresarial*. Obtido em 13 de Outubro de 2009, de <http://www.aurelio.pro.br/computacao/erp.pdf>

Yumi, E. I. (2008). *Movimento Open Source*. Obtido em 11 de Agosto de 2009, de http://www.dicas-l.com.br/download/movimento_open_source.pdf

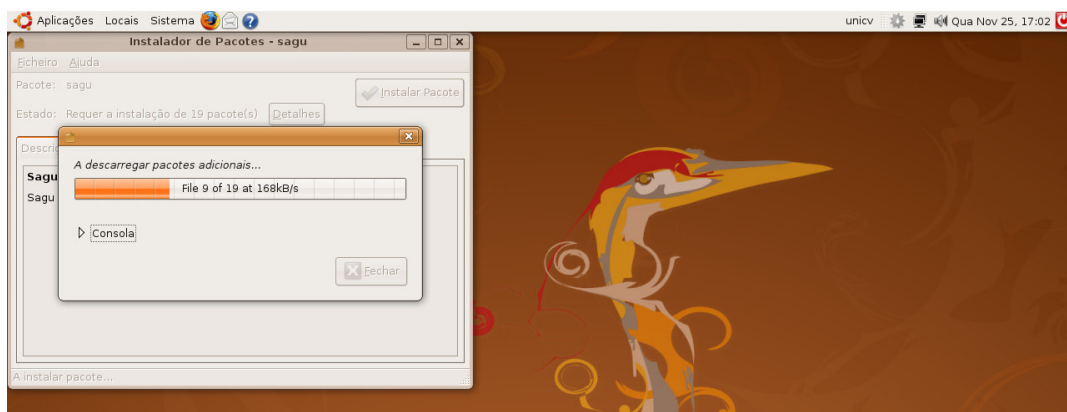
Anexos

Instalação do Sistema

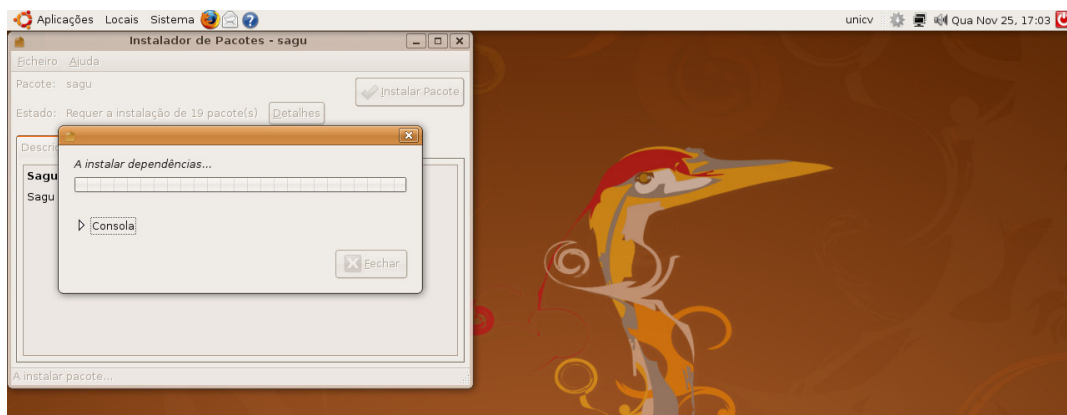
A seguir são descritos os procedimentos necessários para a instalação do SAGU nas mais diversas distribuições GNU/Linux existentes e as configurações iniciais do sistema.

Instalação:

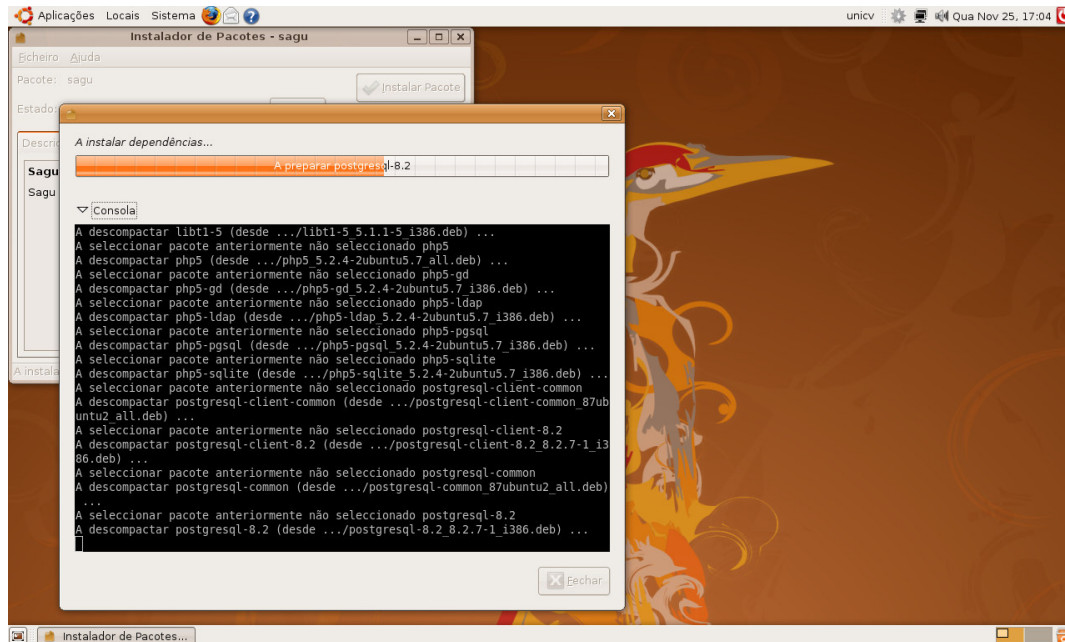
Para a instalação do SAGU, primeiro foi instalado o sistema operativo Ubuntu 8.04 e em seguida a actualizações dos pacotes do sistema.



A instalação pode ser efectuada em modo de consola através de comandos ou em modo interface. Neste trabalho o sistema foi instalado no modo interface através do menu sistema => Gestor de Pacotes Synaptic, que permite a instalação e actualização de todos os pacotes de *softwares*.



O SAGU requer a instalação, prévia do Apache, PostgreSQL e PHP, que já se encontra embutidos no pacote de instalação do SAGU2.



Configurações

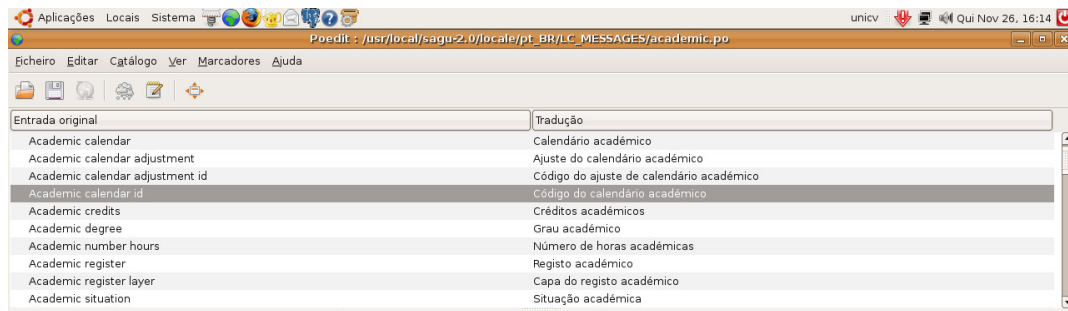
Tradução

Após a instalação do sistema, foi feita a tradução o sistema do Inglês para português. Assim foram efectuadas as seguintes configurações.

É necessário adicionar a codificação "ISO-8859-1" e criar os *locales* do sistema:

```
# echo "pt_BR.ISO-8859-1 ISO-8859-1" >> /var/lib/locales/supported.d/pt
# locale-gen
# localedef -i pt_BR -c -f ISO-8859-1 pt_BR
```

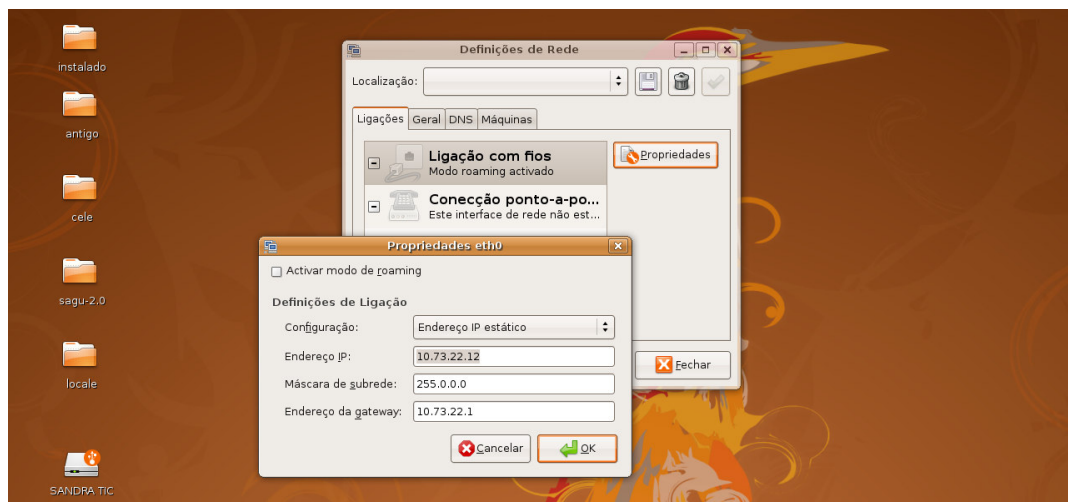
Para a correcção do Português, foi utilizado a ferramenta Poedit.



Colocação do sistema na rede

Após a instalação e a tradução foi colocado o sistema na rede. Para isso procedeu-se os seguintes passos:

- **Configuração do endereço IP Estático;**



- **Configuração do ficheiro sagu2 no directório /etc/apache2/sites-available;**

Para efectuar esta configuração criou-se um VirtualHost e ServerName com o endereço IP que foi configurado estaticamente no servidor.

<VirtualHost 10.73.22.71:80>

ServerAdmin sagu2@solis.coop.br

DocumentRoot /usr/local/sagu-2.0/html

ServerName 10.73.22.71/

</VirtualHost>

<Directory "/usr/local/sagu-2.0/html">

AllowOverride None

Options FollowSymLinks

Order allow,deny

Allow from all

</Directory>

Como foi criado um VirtualHost diferente de "sagu2local", foi configurado o arquivo /usr/local/sagu-2.0/etc/miolo.conf.

3. Configuração do ficheiro miolo.conf do SAGU2;

Para completar a configuração, foi introduzido o endereço IP configurado anteriormente no ficheiro miolo.conf, para que possamos ter acesso ao sistema através da URL.

Parte do código utilizado:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

```
<conf>
```

```
  <home>
```

```
    <miolo>/usr/local/sagu-2.0</miolo>
```

```
    <classes>/usr/local/sagu-2.0/classes</classes>
```

```
    <modules>/usr/local/sagu-2.0/modules</modules>
```

```
    <etc>/usr/local/sagu-2.0/etc</etc>
```

```
    <logs>/usr/local/sagu-2.0/var/log</logs>
```

```
    <trace>/usr/local/sagu-2.0/var/trace</trace>
```

```
    <db>/usr/local/sagu-2.0/var/db</db>
```

```
<html>/usr/local/sagu-2.0/html</html>
<themes>/usr/local/sagu-2.0/classes/ui/themes</themes>
<extensions>/usr/local/sagu-2.0/extensions</extensions>
<reports>/usr/local/sagu-2.0/var/reports</reports>
<images>/usr/local/sagu-2.0/ui/images</images>
<url>http://10.73.22.71</url>
<url_themes>/themes</url_themes>
<url_reports>/reports</url_reports>
<module.themes>/ui/themes</module.themes>
<module.html>/html</module.html>
<module.images>/html/images</module.images>
</home>
```

Dicionário de Dados

Esta sessão faz a descrição das classes constituintes do módulo financeiro.

Nome Físico => representa o atributo, nome único dentro de uma classe.

Tipos de Dados => representam o tipo dados por ex. (Integer, Char, etc).

Necessário => significa se campo é Nulo (*Null*) ou Não Nulo (*Not Null*).

Chave => Atributo identificadora da classe, por ex (PK, FK).

Nota => Descrição das tabelas.

A expressão <PK> significa que o atributo faz parte da chave primária (*primary key*) e a expressão <FK> significa que o atributo faz parte da chave estrangeira (*foreign key*).

Contém comentários referentes a acordos:

Finagreementcomments				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Agreementcommentssid	Integer	Sim	PK	Chave primária da tabela
Personid	Integer	Sim	FK	Chave estrangeira da BasPerson
Agreementtitle	Char (60)	Não		Título do comentário
Comments	Text	Não		Comentários sobre títulos com acordos (acessado pela tela de acordos para pessoas - personAgreements)

Entidade representante dos bancos com quais a instituição trabalha:

Finbank				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Bankid	Char (3)	Sim	PK	Número do banco fornecido pelo Banco Central para manipulação de arquivos
Description	Text	Sim		Nome do banco
Bankidvd	Char (1)	Não		Dígito verificador do número do banco
Accordcode	Char	Não		Código de convênio da empresa com o banco

Entidade representante das contas bancárias:

Finbankaccount				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Bankaccountid	Integer	Sim	PK	Código da conta bancária no SAGU
Description	Text	Sim		Descrição da conta bancária
Accountnumber	Char (30)	Sim		Número da conta bancária
Accountnumberdigit	Char (2)	Não		Dígito verificador do número da conta
Branchnumber	Char (30)	Sim		Número da agência
Branchnumberdigit	Char (2)	Não		Dígito verificador da agência
Bankid	Char (3)	Não	FK	Código do banco (finbank)
Assignorcode	Char (40)	Não		Numero definido pelo banco para composição dos boletos e arquivos de remessa
Wallet	Char (3)	Não		Carteira: utilizado na remessa de arquivos bancários

Entidade representante da movimentação de contas correntes – arquivo do banco (Extracto de conta):

Finbankaccountmovement				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Bankaccountid	Integer	Sim	PK, FK	Código da conta bancária (finbankaccount)
Datemovement	Date	Sim		Data de movimentação do valor actual
Value	Numeric (14,4)	Sim		Valor da movimentação
Operation	Char (1)	Sim		Credito (C) ou Debito (D)

Entidade representante dos débitos automáticos herdadas da entidade fininvoicetarget:

Finbanktarget				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Contractid	Integer	Sim	PK	Código do contrato para identificar a pessoa
Isactive	Text	Sim		Esta flag define se esta ou não activa o destino da cobrança para este contrato, a flag será falsa quando um contrato for desactivado
Bankid	Char (30)	Sim	FK	Código do banco (finbank)
Branchnumber	Char (2)	Sim		Número da agência
Branchnumberdigit	Char (30)	Sim		Dígito verificador da agência
Accountnumber	Char (2)	Sim		Número da conta corrente
Accountnumberdigit	Char (3)	Sim		Dígito verificador da conta corrente

Entidade representante de fechamento de caixa:

Finclosecounter				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Closecounterid	Integer	Sim	PK	Código do fechamento de caixa
Operatorid	Integer	Sim	FK	Funcionário responsável pelo Caixa

Value	Numeric (14,2)	Sim		Valor de fechamento
Registerdate	Date	Sim		Data do registo do caixa
Operation	Char (1)	Sim		Credito (C) ou Debito (D)
Counterid	Integer	Sim	FK	Código do caixa (finCounter)
Isclosed	Boolean	Sim		Define se o fechamento de caixa e parcial ou total

Entidade representante dos tipos de cobranças, podendo ser simples, com registo ou outros:

Fincollectiontype				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Collectiontypeid	Integer	Sim	PK	Código identificador do tipo de cobrança
Description	Text	Sim		Texto descrevendo o tipo da cobrança (simples, com registo, etc)

Entidade representante de Caixas:

FinCounter				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Counterid	Integer	Sim	PK	Código que identifica a caixa
Responsableid	Integer	Sim	FK	Funcionário responsável pela caixa
Unitid	Integer	Sim	FK	Campus onde a caixa de encontra
Iactive	Boolean	Sim		Define se a caixa esta activa ou não

Entidade representante de movimentação da caixa:

Fincountermovement				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Countermovementid	Integer	Sim	PK	Codigo identificador da movimentação de caixa

Counterid	Integer	Sim	FK	Codigo identificador da movimentação de caixa
Operatorid	Integer	Sim	FK	Funcionário operador do caixa
Value	Numeric (14,4)	Sim		Valor da movimentação
Movementdate	Timestamp	Sim		Data e horário da movimentação do caixa
Operation	Char (1)	Sim		Credito (C) ou Débito (D)
Speciesid	Integer	Sim	FK	Espécie monetária (finSpecies)

Operações padrão

Findefaultoperations				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Addcurricularcomponentoperation	Integer	Não	FK	Operação para acréscimos de disciplina
Cancelcurricularcomponentoperation	Integer	Não	FK	Operação para retirada de disciplinas (Credito)
Protocoloperation	Integer	Não	FK	Operação para pagamento de protocolos
Interestoperation	Integer	Não	FK	Operação para juros
Discountoperation	Integer	Não	FK	Operação para descontos
Libraryfineoperation	Integer	Não	FK	Operação para multas da biblioteca
Closeincomeforecastoperation	Integer	Não	FK	Operação para fechamento de previsões
Enrolloperation	Integer	Não	FK	Operação para geração de mensalidades
Paymentoperation	Integer	Não	FK	Operação para pagamento de títulos (fechamento)
Agreementoperation	Integer	Não	FK	Operação para acordos amigáveis
Banktaxoperation	Integer	Não	FK	Operação para operação de taxas bancárias
Selectiveprocesstaxoperation	Integer	Não	FK	Operação utilizada para cobrança de processos selectivos
Bankclosingtaxoperation	Integer	Não	FK	Operação para Fechamento de taxas bancárias

Entidade representante das taxas que podem ser cobradas no acto da matrícula:

Finenrollfee				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota

Learningperiodid	Integer	Sim	PK	Codigo identificador do período lectivo
Operationid	Integer	Sim	PK	Codigo identificador da operação (finOperation)
Isfreshman	Boolean	Sim	PK	Se o aluno e calouro ou não
Valueispercent	Boolean	Sim		Indica se o campo value corresponde a um valor fixo (FALSE) ou a um percentual (TRUE)
Value	Numeric (14,4)	Sim		Se for um valor fixo (ValueIsPercent = FALSE), corresponde ao valor da taxa. Caso contrário, e um percentual
Parcelsnumber	Integer	Sim		Numero de parcelas onde a taxa será cobrada

Entidade que contem os lançamentos dos títulos, tanto dos a receber quanto dos a pagar:

Finentry				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Entryid	Integer	Sim	PK	Codigo identificador do lançamento
Invoiceid	Integer	Sim	FK	Codigo identificador do titulo ao qual o lançamento pertence
Operationid	Integer	Sim	FK	Codigo identificador da operação (finOperation)
Entrydate	Date	Sim		Data de criação do lançamento
Value	Numeric (14,4)	Sim		Valor do lançamento. Somados, os valores dos lançamentos de um título resultam no valor líquido do título
Costcenterid	Char (30)	Sim	FK	Codigo identificador do centro de custo ao qual o lançamento pertence. Este campo e necessário para casos onde a mensalidade e originada de um centro de custo e o desconto referente a sua bolsa de estudos vem de outro centro de custo
Comments	Text	Não		Informação complementar que se deseja adicionar ao lançamento
Bankreturncode	Char (50)	Não		Codigo fornecido pelo banco, geralmente identificando o lote pelo qual o lançamento veio
Isaccounted	Boolean	Sim		TRUE (FALSE) indica que o lançamento (não) foi contabilizado
Creationtype	Char (1)	Não		Tipo de criação: A - Automática, M - Manual

Entidade para registo de arquivos de retorno, remessa e outros arquivos que são processados e enviados para o servidor:

Finfile				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Fileid	Integer	Sim	PK	Código Identificador do arquivo
Filename	Text	Sim		Nome do arquivo
Bankaccountid	Integer	Não		Conta bancária a qual esta ligada o arquivo (finbankaccount)
Filetype	Char (1)	Sim		Define o tipo de arquivo de conversação I = invoice (título), A =Automatic debit, M = Messages, T = Tax
Inputoroutput	Char (1)	Sim		I) Para INPUT - (Retorno), (O) para OUTPUT - (Remessa)
Isprocessed	Boolean	Sim		Define se o arquivo foi processado ou não

Entidade para incentivos do tipo bolsa de estudo:

Finfinancialaid				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incentivetypeid	Integer	Sim	PK	Código identificador do tipo de incentivo
Description	Text	Sim		Texto descrevendo o tipo de incentivo
Operationid	Integer	Não		Código identificador da operação do incentivo (finOperation)
Needadjustauthorization	Boolean	Sim		Se o aluno precisa autorização da empresa para fazer ajuste de matrícula
Sendinvoices	Boolean	Sim		Envia ou não títulos de pessoas que possuem este tipo de incentivo
Isexpired	Boolean	Sim		Se um incentivo foi ou não cancelado
Generatecredits	Boolean	Não		Gerar créditos
Requirecostcenter	Boolean	Sim		Obrigar o preenchimento de costCenter em finIncentive

Entidade para incentivos de estudo dados a alunos, como bolsas, financiamentos, patrocínios, etc:

FinIncentive				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incentiveid	Integer	Sim	PK	Codigo identificador do incentivo
Contractid	Integer	Sim	FK	Codigo identificador do contrato (pessoa) ao qual o incentivo será concedido
Startdate	Date	Sim		Data a partir da qual o incentivo passa a valer
Enddate	Date	Sim		Data de expiração do incentivo
Incentivetypeid	Integer	Sim	FK	Codigo identificador do tipo de incentivo (finIncentiveType)
ValueIsPercent	Boolean	Não		TRUE (FALSE) se o campo value representa um percentual (valor)
Value	Numeric (14,4)	Não		Representa o valor (ValueIsPercent = FALSE) ou o percentual (ValueIsPercent = TRUE) do desconto sobre os títulos de mensalidade emitidos
Supporterid	Integer	Sim	FK	Codigo identificador da pessoa que será o patrocinador do incentivo. Só é preenchido se for um patrocínio
Agglutinate	Boolean	Sim		TRUE se o financiador receberá apenas um título com o valor total dos Incentivos que financia ou FALSE se deverá ser emitido um título para cada Incentivo financiado
Costcenterid	Char (30)	Não	FK	Codigo identificador do centro de custo ao qual o incentivo pertence
Cancellationdate	Date	Não		Indica quando o incentivo foi cancelado (caso finalizado antes do previsto)

Entidade tipos de incentivos que podem ser concedidos:

FinIncentiveType				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incentivetypeid	Integer	Sim	PK	Codigo identificador do tipo de incentivo
Description	Text	Sim		Texto descrevendo o tipo de incentivo
Operationid	Integer	Não	FK	Codigo identificador da operação do incentivo (finOperation)
Needadjustauthorization	Boolean	Sim		Se o aluno precisa autorização da empresa para fazer ajuste de matrícula

Sendinvoices	Boolean	Sim		Envia ou não títulos de pessoas que possuem este tipo de incentivo. (FAE)
Isexting	Boolean	Sim		Envia ou não títulos de pessoas que possuem este tipo de incentivo. (FAE)
Generatecredits	Boolean	Não		Gerar crédito

Entidades contem as previsões de lançamento necessárias para geração automática dos títulos de matrículas e processos cujo pagamento ocorre em mais de uma parcela.

Finincomeforecast				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incomeforecastid	Integer	Sim	PK	Código identificador da previsão de lançamento
Contractid	Integer	Sim	FK	Código identificador do contrato (pessoa) ao qual a previsão pertence
Operationid	Integer	Não	FK	Código identificador da operação (finOperation)
Accountschemeid	Char (30)	Sim		Código identificador da conta contábil a qual a previsão pertence
Costcenterid	Char (30)	Sim	FK	Código identificador do centro de custo ao qual a previsão pertence
Value	Numeric (14,4)	Sim		Valor da previsão de lançamento
Comments	Text	Não		Informação complementar da previsão de lançamento
Recorddate	Date	Não		Data de criação de previsão
Maturitydate	Date	Não		Data da criação da previsão
Isaccounted	Boolean	Sim		Data em que a previsão devera se tornar título
Isprocessed	Boolean	Sim		TRUE (FALSE) indica que a previsão (não) foi contabilizada
Isgenerated	Boolean	Sim		TRUE (FALSE) indica que a previsão (não) foi transformada em título
Issuppressed	Boolean	Sim		TRUE gerado pelo sistema, FALSE gerado manual
Incentiveid	Integer	Não	FK	Campo indicando de qual incentivo foi gerado esta previsão
Issuppressed	Boolean	Sim		Marca se a previsão e suprimida (Tem que ser contabilizada mas não e gerado lançamento desta)

Enrollsummaryid	Integer	Não	FK	Caso a previsão foi gerada a partir de um resumo de matricula
-----------------	---------	-----	----	---

Entidade origens de receita:

Finincomesource				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incomesourceid	Integer	Sim	PK	Codigo identificador da origem
Description	Text	Sim		Descrição da origem
Isextinct	Integer	Sim		Define se a origem esta extinta ou não
Isagreement	Boolean	Não		Acordo

Entidade principais dos títulos, contendo tantos títulos a pagar/pagos quantos títulos a receber/recebidos:

Fininvoice				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Invoiceid	Integer	Sim	PK	Codigo identificador do título
Personid	Integer	Sim	FK	Codigo identificador da pessoa a qual o titulo pertence
Accountschemeid	Char (30)	Sim	FK	Codigo identificador da conta contabil a qual o titulo pertence
Costcenterid	Char (30)	Sim	FK	Codigo identificador do centro de custo ao qual o titulo pertence
Courseid	Char (30)	Não	FK	Codigo identificador do curso ao qual a pessoa que originou o titulo pertence
Courseversion	Integer	Não		Versão do curso
Unitid	Integer	Não	FK	Codigo identificador do campus ao qual o titulo pertence
Parcelnumber	Integer	Sim		Codigo identificando qual parcela (do total de parcelas) este título representa
Emissiondate	Date	Sim		Data de emissão do título
Maturitydate	Date	Sim		Data de vencimento do título

Value	Numeric (14, 4)	Sim		Valor do título
Policyid	Integer	Sim	FK	Codigo identificador da politica utilizada pelo título (finPolicy)
Bankinvoiceid	Char (30)	Não		Codigo fornecido pelo banco para identificação do título
Automaticdebit	Boolean	Sim		TRUE (FALSE) indica que o título (não) foi quitado via debito automático em conta
Comments	Text	Não		Informações adicionais do título
Incomesourceid	Integer	Sim	FK	Origem de receitas
Bankaccountid	Integer	Sim	FK	Codigo da conta corrente (finbankaccount) que recebera o dinheiro deste pagamento
Sagulinvoiceid	Char (20)	Não		Para não perder a referencia a forma antiga de identificação dos títulos (SAGU1)
Sectorid	Integer	Não	FK	Sector

Log de títulos gerados pelo processo de geração automática de títulos:

Fininvoicelog				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Month	Integer	Sim	PK	Mes de geração
Year	Integer	Sim	PK	Ano de geração
Generationdate	Date	Sim		Data em que o processo de geração de títulos foi disparado
Generationtype	Char (1)	Sim	PK	Indica qual o tipo de geração do processo de títulos

Destino de cobrança:

Fininvoicetarget				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Contractid	Integer	Sim	PK	Codigo do contrato para identificar a pessoa
Isactive	Boolean	Sim		Esta flag define se esta ou não activo o destino da cobrança para este contrato, a flag será falsa quando um contrato for desactivado

Incentivo do tipo financiamento. Herda a finIncentiveType:

Finloan				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incentivetypeid	Integer	Sim	PK	Código identificador do tipo de incentivo
Description	Text	Sim		Texto descrevendo o tipo de incentivo
Operationid	Boolean	Sim		Código identificador da operação do incentivo (finOperation)
Needadjustauthorization	Boolean	Sim		Se o aluno precisa autorização da empresa para fazer ajuste de matrícula
Sendinvoices	Boolean	Sim		Envia ou não títulos de pessoas que possuem este tipo de incentivo. (FAE)
Isextinguish	Boolean	Sim		Se um incentivo foi ou não cancelado
Generatecredits	Boolean	Não		Gerar créditos
Accountschemeid	Char (30)	Sim	FK	Código identificador da conta a contábil qual o financiamento pertence
Loanerid	Integer	Não	FK	Código identificador da pessoa financiadora do incentivo
Rotative	Boolean	Sim		TRUE se o valor patrocinado ao aluno deverá ser pago por ele próprio no futuro. Caso contrário, FALSE
Groupinvoicesvalues	Integer	Sim		TRUE se os valores devem ser agrupados em um único título (caso do PROCRES, por exemplo) ou se devem ser gerados vários títulos (caso do PCR)
Collectionoperationid	Integer	Sim	FK	Código identificador da operação (finOperation) de cobrança

Cadastro de mensagens que poderão compor os boletos bancários, tais como "pagável até o dia x em qualquer agência bancária:

Finmessage				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Bankid	Char (3)	Sim	PK, FK	Código identificador do banco (finbank)
Collectiontypeid	Integer	Sim	PK, FK	Código identificador do tipo de cobrança (fincollectiontype)
Messageorder	Integer	Sim	PK	Ordem em que a mensagem deverá aparecer no folhas,

				sendo a primeira mensagem a de menor ordem
Bankmessagecode	Char (20)	Não		Código identificador da mensagem (para o código de mensagens do banco) - Fornecido pelo banco
MessageText	Text	Não		Texto da mensagem

Identifica uma operação para um determinado ocorrência, utilizado no arquivo de retorno bancário:

Finoccurrenceoperation				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Occurrenceoperationid	Integer	Sim	PK	Código identificador da ocorrência de operação
Occurrenceid	Char (2)	Sim		Código da ocorrência, este dado e padrão bancário (BCN)
Bankaccountid	Integer	Sim	FK	Código da conta bancária (finbankaccount)
Operationid	Integer	Sim	FK	Código identificador da operação (finOperation)
Levelsequenceid	Integer	Sim		Valor do nível da sequência, utilizado para adquirir uma determinada operação

Identificam a natureza de um lançamento ou previsão (débito ou crédito):

FinOperation				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Operationid	Integer	Sim	PK	Código identificador da operação
Description	Text	Sim		Texto descrevendo a operação
Operationtypeid	Boolean	Sim		Debito (D) ou credito (C)
Isexting	Boolean	Sim		Define se uma operação esta extinta ou não
Iscountermovement	Boolean	Sim		Se essa operação terá ou não movimentação de caixa
Operationgroupid	Char (1)	Sim		Código identificador do grupo da operação

Define o grupo de operação (pagamento, incentivo, juros, descontos, normal):

Finoperationgroup				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Operationgroupid	Char (1)	Sim	PK	Código identificador do grupo da operação
Description	Text	Sim		Texto descrevendo o grupo da operação

Entidade contendo as informações adicionais a tabela fininvoice sobre títulos a pagar ou pagos:

Finpayableinvoice				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Invoiceid	Integer	Sim	PK	Código identificador do título
Personid	Integer	Sim		Código identificador da pessoa a qual o título pertence
Accountschemeid	Char (30)	Sim		Código identificador da conta contábil a qual o título pertence
Costcenterid	Char (30)	Sim		Código identificador do centro de custo ao qual o título pertence
Courseid	Char (10)	Sim		Código identificador do curso ao qual a pessoa que originou o título pertence
Courseversion	Integer	Não		Versão do curso
Unitid	Integer	Não		Código identificador do campus ao qual o título pertence
Parcelnumber	Integer	Sim		Código identificando qual parcela (do total de parcelas) este título representa
Emissiondate	Date	Não		Data de emissão do título
Maturitydate	Date	Sim		Data de vencimento do título
Value	Numeric (14,4)	Sim		Valor do título
Policyid	Integer	Sim		Código identificador da política utilizada pelo título (finPolicy)
Bankinvoiceid	Char (30)	Não		Código fornecido pelo banco para identificação do título
Automaticdebit	Boolean	Sim		TRUE (FALSE) indica que o título (não) foi quitado via débito automático em conta
Comments	Text	Não		Informações adicionais do título

Incomesourceid	Integer	Sim		Origem de receitas
Bankaccountid	Integer	Sim		Código da conta corrente (finbankaccount) que recebera o dinheiro deste pagamento
Sagulinvoiceid	Char (20)	Não		Para não perder a referencia a forma antiga de identificação dos títulos
Receivedate	Date	Não		Data em que o título foi recebido pela instituição para que a mesma efectuassee seu pagamento

Entidade débito em folha de pagamento (de funcionários) a nível de contrato:

Finpayrolldiscounttarget				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Contractid	Integer	Sim	PK	Código do contrato para identificar a pessoa
Isactive	Boolean	Sim		Esta flag define se esta ou não activa o destino da cobrança para este contrato, a flag será falsa quando um contrato for desactivado
Ispayrolldiscount	Boolean	Sim		Define se o pagamento ao descontado em folha ou não
Employeeid	Integer	Sim	FK	Se tiver um Código de funcionário (personid) neste campo, quer dizer que o dono deste contrato quer que sua mensalidade seja descontada na folha de pagamento deste funcionário. Se não tiver nenhuma informação neste campo o desconto será feito na folha do titular do contrato

Informação financeira de pessoas físicas e jurídicas. (antigo campo observação da tabela pessoas do Sagui):

Finpersoninformation				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Personid	Integer	Sim	PK, FK	Código identificador da pessoa as quais as informações pertencem
Information	Text	Não		Informação da pessoa

Endereço de cobrança. Herda a fininvoicetarget:

Finphysicaltarget

Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Contractid	Integer	Sim	PK	Código do contrato para identificar a pessoa
Isactive	Boolean	Sim		Esta flag define se esta ou não activo o destino da cobrança para este contrato, a flag será falsa quando um contrato for desactivado
Cityid	Integer	Sim	FK	Código identificador da cidade para cobrança
Zipcode	Char (9)	Sim		CEP do endereço de cobrança
Location	Char (100)	Sim		Logradouro
Complement	Char (40)	Não		Complemento
Number	Char (10)	Não		Numero
Neighborhood	Text	Não		Bairro
Name	Char (10)	Sim		Nome da pessoa que deve aparecer no folha
Documentnumber	Char (11)	Não		CPF do nome

Define política e regras de negócio do financeiro:

FinPolicy				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Policyid	Integer	Sim	PK	Código identificador da política
Description	Text	Sim		Descrição da política
Operationid	Integer	Sim	FK	Código identificador da operação padrão para matrículas de cursos que seguem esta política
Collectiontypeid	Integer	Sim	FK	Código identificador do tipo de cobrança (fincollectiontype)
Monthlyinterestpercent	Integer	Sim		Percentual de juros a serem cobrados no mes
Finepercent	Double	Sim		Percentual cobrado em caso de multa
Daystointerest	Integer	Sim		A partir de quantos dias será cobrado juro
Daystofine	Integer	Sim		A partir de quantos dias será cobrada multa
Daystodiscount	Integer	Sim		Dias para desconto
Daystoprotest	Integer	Sim		A partir de quantos dias será protestado o título

Discountpercent	Double	Sim		Percentual de desconto
Banktaxvalue	Numeric (14,4)	Sim		Valor da taxa bancária
Isexting	Boolean	Sim		Define se a politica esta ou não activa
Interesttype	Char (1)	Não		ipo de juros "S: Simples", "C: Composto"
Applyinterest	Boolean	Não		Aplicar regras dos juros mensais para esta política?
Applyfine	Boolean	Não		Aplicar regras de multas para esta política?

Valor do curso para um determinado período:

Finprice				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Learningperiodid	Integer	Sim	PK, FK	Codigo identificador do período lectivo (acdLearningPeriod)
Startdate	Date	Sim	PK, FK	Data de inicio
Enddate	Date	Sim	PK, FK	Data final
Value	Numeric (14,4)	Sim		Preço do curso

Contêm as informações adicionais a tabela fininvoice sobre títulos a receber ou recebidos:

Finreceivableinvoice				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Invoiceid	Integer	Sim	PK, FK	Codigo identificador do título
Personid	Integer	Sim		Codigo identificador da pessoa a qual o titulo pertence
Accountschemeid	Char (30)	Sim		Codigo identificador da conta contabil a qual o titulo pertence
Costcenterid	Char (30)	Sim		Codigo identificador do centro de custo ao qual o titulo pertence
Courseid	Char (10)	Não		Codigo identificador do curso ao qual a pessoa que originou o titulo pertence
Courseversion	Integer	Não		Versão do curso

Unitid	Integer	Não		Codigo identificador do campus ao qual o titulo pertence
Parcelnumber	Integer	Sim		Codigo identificando qual parcela (do total de parcelas) este titulo representa
Emissiondate	Date	Sim		Data de emissão do titulo
Maturitydate	Date	Sim		Data de vencimento do titulo
Value	Numeric (14,4)	Sim		Valor do titulo
Policyid	Integer	Sim		Codigo identificador da politica utilizada pelo titulo (finPolicy)
Bankinvoiceid	Char (10)	Sim		Codigo fornecido pelo banco para identificação do titulo
Automaticdebit	Boolean	Não		TRUE (FALSE) indica que o titulo (não) foi quitado via debito automático em conta
Comments	Text	Não		Informações adicionais do titulo
Incomesourceid	Integer	Sim		Origem de receitas
Bankaccountid	Integer	Sim		Codigo da conta corrente (finbankaccount) que recebera o dinheiro deste pagamento
Sagulinvoiceid	Char (20)	Não		Para não perder a referência a forma antiga de identificação dos títulos (SAGU2)
Sectorid	Integer	Não		Sector
Senddate	Date	Não		Data em que o titulo foi enviado para o banco
Returndate	Date	Não		Data em que o titulo retornou do banco
<u>Returnfileid</u>	Integer	Não	FK	Arquivo de retorno do banco

Entidade informações sobre comunicações de arquivos com bancos.

Finreceivableinvoicecommunication				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Receivableinvoicecommunicationid	Integer	Sim	PK	Chave primária da classe
Invoiceid	Integer	Não	FK	Codigo do titulo referente a comunicação
Fileid	Integer	Não	FK	Codigo do arquivo ao qual foi efectuada a comunicação
Commdate	Date	Não		Data da comunicação do arquivo

Espécies monetárias ex. cheque, dinheiro.

Finspecies				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Speciesid	Integer	Sim	PK	Código identificador da espécie
Description	Integer	Não		Descrição da espécie

Financiamento estudantil (fies).

Finstudentfinancing				
Nome Físico	Tipos de Dados	Necessário	Chave Primária	Nota
Studentfinancingid	Integer	Sim	PK	Código identificador do financiamento estudantil
Contractid	Integer	Sim	FK	Contrato do aluno (acdContract)
Initialvalue	Numeric (14,4)	Sim		Valor inicial do financiamento
Currentvalue	Numeric (14,4)	Sim	FK	Valor actual do financiamento
Comments	Text	Não		Observações

Incentivo do tipo patrocínio (empresas que custeiam os estudos dos funcionários, por exemplo). Herda a finincentivetype.

Finsupport				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Incentivetypeid	Integer	Sim	PK	Código identificador do tipo de incentivo
Description	Text	Sim		Texto descrevendo o tipo de incentivo
Operationid	Integer	Sim		Código identificador da operação do incentivo (finOperation)
Needadjustauthorization	Boolean	Sim		se o aluno precisa autorização da empresa para fazer ajuste de matrícula
Sendinvoices	Boolean	Sim		Envia ou não títulos de pessoas que possuem este tipo de incentivo

Isexpired	Boolean	Sim		e um incentivo foi ou não cancelado
Generatecredits	Boolean	Não		Gerar créditos
Accountschemeid	Char (30)	Sim	FK	Código identificador da conta contábil a qual o patrocínio pertence
Collectionoperationid	Integer	Sim	FK	Operação para a cobrança para o patrocinador

Entidade para mensagens para os demonstrativos financeiros, comprovantes genéricos:

Finvouchermessages				
Nome Físico	Tipos de Dados	Necessário	Chave	Nota
Operationid	Integer	Sim	PK	Código identificador da operação
Title	Text	Sim	Não	Título do demonstrativo
Message	Text	Sim	Não	Mensagem do demonstrativo